AD-A259 142

AFIT/GCS/ENG/92D-09

A UNIFIED MODEL OF PROGRAM BEHAVIOR

THESIS

Douglas T. Michel
Captain, USAF

AFIT/GCS/ENG/92D-09

DTIC
ELECTE
JAN 1 1 1993
S E D

012225
93-00087
169 pp

Approved for public release; distribution unlimited

93 1 4 057

AFIT/GCS/ENG/92D-09

A UNIFIED MODEL OF PROGRAM BEHAVIOR

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Computer Engineering

DTIC QUALITY INSPECTED 5

Accesion For		
NTIS CRA&I	X	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

Douglas T. Michel, B.S.C.S.

Captain, USAF

Dec, 1992

Approved for public release; distribution unlimited

Acknowledgements

I would like to thank my thesis advisor LtCol William C. Hobart, Jr. for the guidance and encouragement throughout the writing of this thesis. I would also like to thank my committee members, Dr Bruce Suter and Maj Tom Wailes for their comments and suggestions.

Douglas T. Michel

Table of Contents

Page

List of Figures

Figure		Page

List of Tables

Table	Page

AFIT/GCS/ENG/92D-09

Abstract

This thesis is an attempt to account for and unify the three types of locality: temporal, spatial, and structural. A diverse sample of traces are used in measuring program behavior with respect to these localities and a model is presented which represents the memory references a program generates as it goes through execution. The model is validated by estimating the entropy of a synthetically generated trace and comparing it with actual traces. The results indicate that there is more predictability contained in the original trace than what the model was able to capture. Different variations of the model were tried and the results varied depending on the trace type being modeled. Various other measurements concerning temporal, spatial, and structural locality are used in building the model and provide interesting and useful insight into the memory referencing patterns of programs.

A UNIFIED MODEL OF PROGRAM BEHAVIOR

I. Introduction

1.1 Background

As a computer program is executed it generates a sequence of references to memory. These references are for retrieving the instructions of the program stored in memory, to load or read data required by the instructions, and to store or write data as results are computed. This sequence of memory references is a factor in the performance of a computer system. References which result in long memory access times result in decreased performance; conversely, references with short memory access times result in enhanced performance. This referencing of memory by a program is known as program behavior.

Program behavior can be characterized by the localities the program exhibits as it goes through execution. The property of locality can be informally defined as the tendency of a program to favor a smaller portion of its memory space allowing a program to execute efficiently without all of it in memory.

Locality has typically been categorized as being spatial or temporal. Spatial locality is the tendency of a program's memory references to be located 'close' to one another, while temporal locality is the tendency of a memory reference to be referenced again within a 'short' period of time. A third type of locality of reference known as structural locality, introduced by Thazhuthaveetil [TP87] and further defined by Hobart [Hob89], is the tendency of a program to re-reference memory in the same order in which it was previously referenced. These types of localities are formally defined in Section 1.3. Models of program behavior have typically relied on the temporal aspects of locality, while the spatial aspects have been exploited through techniques such as prefetching.

The temporal and spatial aspects of locality are exploited throughout all levels of the computer system's memory hierarchy. The memory hierarchy consists of slow, inexpensive, massive storage devices at the lowest levels farthest from the CPU. As the hierarchy progresses upward, closer to the CPU, these storage devices are faster. This speed results in these storage devices being more expensive and less massive. For the purposes of this research, the lowest level of the memory hierarchy is main memory. At this level, memory locations are grouped into pages. Program behavior at the page level has been extensively researched to find the optimal memory management policy resulting in increased performance. Various models have been proposed to increase performance with the best known being Denning's Working Set Model [Den68]. At a higher level in the memory hierarchy, performance has also been enhanced through the use of caches. Memory locations at this level are grouped into blocks with the smallest possible block being a single memory location or word. Caches are smaller and faster than main memory and are the first level of memory the computer references. Considerable research has been conducted on caches and how their various parameters affect computer performance. The success at both levels in the improved performance of computers has been due to the naturally occurring phenomena of locality which is inherent (to some degree) in all programs.

While program behavior and the locality that is exhibited by programs has been modeled at the main memory page level, program behavior at the memory address reference level of caches is still a relatively new domain. There is no generally accepted model for program behavior [Smi87]. The success of managing memory depends on the ability of the underlying model upon which the memory management system is based to predict the program's referencing behavior.

1.2 Problem Statement

The purpose of this thesis is to provide a general model of program referencing behavior which accounts for and unifies the spatial, temporal, and structural localities exhibited by a program and

demonstrates the relationship of these localities to each other. This model can help explain the various effects a given parameter can have on the performance of a cache design and provide a better understanding of and insight into the memory referencing behavior of programs.

1.3 Definitions

1.3.1 Locality of Reference is a term used to describe a program's preferred set of referenced memory locations. References which are in the current locality have a higher probability of being referenced than those that are not. Denning described the following three characteristics of locality of reference [DS72]:

1. The set of memory references for a given program are distributed in a nonuniform fashion.
2. A program's tendency to reference a given memory location changes slowly over time.
3. The correlation between any two segments of a program's memory reference trace is high when the interval between the segments is small. This correlation goes toward zero as the interval becomes large.

1.3.2 Temporal Locality is a locality of reference in which a reference to a given location implies that the same location will be referenced again in the near future. Temporal locality is exhibited in loop structures during which the same instructions and data are referenced repeatedly.

1.3.3 Spatial Locality is a locality of reference in which a reference to a given location implies that future references are located close to that location in the memory address space. Spatial locality is exhibited through the inherent sequential nature of programs.

1.3.4 Structural Locality is a locality of reference in which references to a sequence of locations implies that the same sequence will be re-referenced. Structural locality is exhibited by branches and subprogram calls.

1.4 Thesis Structure

The next chapter reviews previous research as it applies to this thesis effort. Definitions of terms related to the study of program behavior and the memory hierarchy are given, and models of program behavior are summarized as well as approaches to characterize and measure locality. Next, analytical cache models which have been developed and studied to characterize cache performance when executing a program are examined, and the characterization and research of structural locality is explored. Finally, techniques used to derive the sequences of actual program memory address references known as traces which are used in this research are discussed along with cache simulation studies investigating the effects of locality on cache performance.

Chapter 3 provides the methodology and approach used to build this general model of program behavior. Characteristics of these traces with respect to spatial, temporal, and structural locality are summarized. A unified model is then developed and refined based on the locality characteristics of the traces.

Chapter 4 discusses the results of this research and the validity of the model. Temporal, spatial, and structural locality aspects of program traces are given and compared to the model's prediction of these locality aspects.

Chapter 5 describes the conclusions that can be drawn from this research and gives recommendations for areas in need of further investigation.

II. Background

This chapter reviews previous work in program behavior. It begins with an explanation of some additional terms that are used in this research followed by an examination of early models of program behavior and a discussion of the Least Recently Used Stack Model (LRUSM). The phases and transitions of programs exploited by the Bounded Locality Interval (BLI) Model and its various derivatives are discussed. Other studies of program behavior and locality are reviewed followed by a discussion of analytical cache models which also have been used to help explain program behavior. Research into structural locality is then discussed. Finally, the traces used in this research and the way in which they are derived is surveyed.

2.1 Memory Management Terms

The memory structure of a computer system can be described as a hierarchy. The memory with the most capacity and slowest access time (main memory) is located at the lowest level, and the memory with smallest capacity and fastest access time (cache) is located at the highest level which is closest to the CPU. [1] As a computer executes a program, it references instructions and data from memory. Data may be either read from or written to memory. Programs are typically located on a secondary storage device such as a disk drive. As the program is executed sets of references are loaded into the appropriate level of the memory hierarchy as they are needed. The property of locality described earlier permits a program to execute without being entirely in main memory. Efficient management of memory at all levels is important to the performance of a computer system. Before defining various terms, the concept of virtual memory should be discussed. Virtual memory allows programs to logically exist in an address space which does not physically exist. When the

[1] In an actual computer system secondary or long term storage exists in a level beneath main memory. This level has larger storage capability, is much slower, and is implemented with technologies such as magnetic disk or drum. Because these levels are not as important to the virtual address space, it is not considered in this research.

contents of an address are required by the program, the virtual address is mapped to a physical address in memory. The following terms, used in memory management, are discussed in this thesis.

Virtual Address - The logical address of an item before the address is mapped into physical (real) memory.

Page - The set of memory address references constituting one segment of main memory.

Page fault (page exception) - Caused by the occurrence of a reference to a page not in main memory, resulting in an access to secondary storage. It results in the newly referenced page being loaded into main memory.

Page fault rate - The number of memory references to pages not found in memory divided by the total number of memory references.

Line (block) - The set of memory address references constituting one segment of a cache.

Hit - A memory reference that is present in the cache.

Miss - A memory reference that is absent from the cache.

Miss ratio - The number of misses divided by the total number of references to the cache.

Average access time - The average amount of time it takes to retrieve data from memory to the CPU.

Pollution - The portion of a block's contents which have not been referenced.

Associativity - Determines how a cache is organized and where a block may be placed in the cache. In a *set associative* cache, a block is mapped to a set and the block is placed anywhere within that set. *N-way set associativity* means that there are n blocks in the set. In a one-way set associative cache, a given block can be placed in only one location in the cache. This organization is also known as *direct mapped*. An organization which allows a block to be placed anywhere in the cache is called *fully associative*. In a cache which is fully associative and contains m blocks, the cache can also be called *m-way set associative* [HP90].

Trace - A sequence of memory address references recorded from an actual program as it is executed and denoted by:

$$R_0, R_1, R_1, \ldots$$

where R_i is the address referenced at time i. Traces are used in trace-driven simulations to simulate a program's performance for a given memory design. Traces consist of memory address references due to instruction fetches, data reads, and data writes and are also known as reference strings.

Prefetching - The storage of the contents of memory addresses which have not yet been requested by the CPU in a higher level of the memory hierarchy.

Replacement policy - Decides which block or page to replace when that level of the memory hierarchy becomes full in order to make room for the newly referenced block/page retrieved from a lower level. Replacement policies typically exploit temporal locality. A discussion of various replacement policies is provided by Belady in [Bel66].

2.2 Early Program Behavior Models

One the earliest models of program behavior, which is still studied and used today, is Denning's working-set model [Den68]. The working-set can be defined as the set of objects (pages/blocks) which are needed to assure a certain level of processing efficiency. This working-set changes over time and is denoted by $W(t, \tau)$, where t is a specified time and τ is a time interval known as the working-set parameter. Properties of the working-set are size (working-set size), prediction (past page references to predict future references), reentry rate (referenced page/block not in the existing working-set), and τ-Sensitivity (sensitivity of reentry rate to changes in τ). This model depends upon the temporal locality of the program since references that have been made in the past (working-set at $t = t_1$ say) are likely to be referenced in the future (working-set at $t = t_2$).

Chu and Opderbeck make the point that an 'ideal' replacement algorithm should not make use of prior knowledge of program behavior and that information regarding replacement of pages and efficient memory allocation be gathered while the program is executing [CO76]. Their page fault frequency (PFF) algorithm uses the page-fault frequency to determine how best to allocate pages in memory. When the page-fault frequency rises above some critical page-fault frequency level (P), all referenced pages causing page faults are brought into main memory without replacing any pages resulting in an increased number of allocated page frames. Comparisons were made between the PFF algorithm and the Least Recently Used (LRU) replacement algorithm and their results indicate that PFF is better than LRU replacement. An analytical model for the PFF algorithm is discussed and the authors show that it is a good predictor of performance. One of the problems with the PFF replacement algorithm is that non-referenced pages can stay in memory when the interpage-fault time is long resulting in inefficient operation. To overcome this, Chu and Opderbeck suggest that when the interpage-fault time exceeds some maximum interpage-fault time, pages which have not been referenced since the last page fault are released. They point out that the minimum number of page faults associated with any replacement strategy is equal to number of unique pages referenced by the program during execution. Like the working-set, the PFF replacement policy is dynamic and allows the number of allocated page frames devoted to a given process to shrink and grow.

Lewis and Shedler present a two-state Markov model in an attempt to model the occurrence of page faults [LS73]. One state represents frequent occurrence of page faults while the other represents infrequent page faults. They were reasonably successful in fitting the model to data obtained from trace samples. They suggest that a third state be added to account for behavior during which long periods of referencing a page result in page faults which only occur at the edge of a page. They note that their model is a micromodel since it assumes that transition probabilities do not change over time (stationarity). Spirn states that a micromodel is concerned only with program behavior within a phase, whereas a macromodel takes into account phase transitions with each macro state corresponding to a micromodel [Spi76].

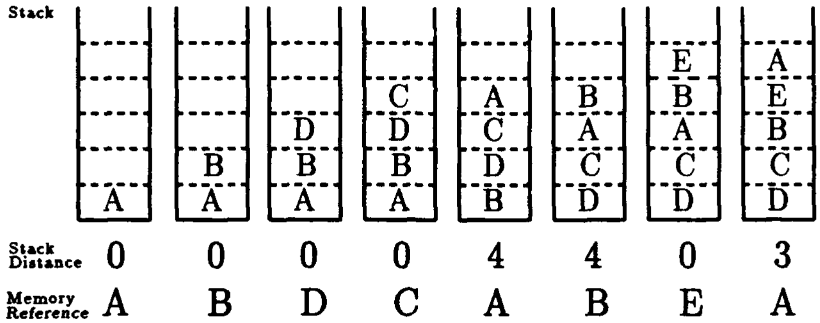

Figure 2.1. Sequence of LRU Stacks.

2.3 LRU Stack Model

Mattson and others discuss various stack oriented algorithms as an approach to modeling program behavior [MGST70]. The most popular and promising of these approaches has been to use a Least Recently Used (LRU) stack to arrange the sequence of memory references of a program according to usage. The most recently referenced address is located at the top of the stack, while the least recently referenced address at the bottom. Instead of using memory addresses, Spirn and Denning use stack distances in their model [SD72]. In the string of stack distances, large stack distances imply that a memory location has not been referenced for a long time while small stack distances imply that a memory location has been referenced in the recent past. A stack distance of zero denotes a previously unreferenced memory location. Figure 2.1 illustrates how a sequence of memory address references would appear on the stack and what the corresponding stack distance would be. As new references are encountered they are 'pushed' onto the stack. If an old reference is encountered it is 'pulled' from its position in the stack a put on top of the stack. The rest of the stack is reordered appropriately.

Spirn and Denning classify models of program behavior as being either intrinsic or extrinsic, and they investigated the ability of several intrinsic models to estimate locality. Intrinsic models define locality to be dependent upon the internal state of the program. Extrinsic models "define locality in terms of observable properties of the memory reference sequence of the program." Ex-

trinsic models are used to estimate intrinsic localities and predict future intrinsic localities. The intrinsic models they studied were the Very Simple Locality Model (VSLM), the Simple LRU Stack Model (SLRUM), and the Independent Reference Model (IRM). The working-set model is used as an extrinsic model to help estimate locality. They find stack models to be good models of program behavior.

The LRU stack model is further investigated by Spirn in [Spi76]. Spirn's model of program behavior uses LRU distance strings which characterize of the temporal locality in a program. The LRU stack model is developed by assigning probabilities to each of the stack distances with increasing distances having a lower probability of being referenced. The LRU stack model does not take into account varying sizes in the working-set. In other words, the working-set is a dynamic memory allocating scheme whereas LRU is fixed.

2.4 Phases, Transitions, and the BLI Model

By using LRUSM, Madison and Batson characterize locality by using the concept of Bounded Locality Intervals (BLIs) to describe localities in a program [MB76]. They key in on the phase-transition behavior of programs where a program will be in a certain phase of execution during which references are to a favored set of references. Transitions are periods during which this favored set of locations would change after which another phase would be entered. Like the working-set, BLIs are defined as a 2-tuple (A_i, τ_i) where A_i is the activity set and τ_i is its lifetime. BLIs can contain BLIs, thus, they are hierarchical in nature. BLIs with long lifetimes denote the phases of a program whereas short-lived BLIs denote transitions between phases.

Batson, Blatt, and Kearns use BLIs to focus on the cyclic patterns found in loops and iteration structures [BBK77]. Locality intervals with this kind of characteristic are known as cyclic locality intervals. Cyclic locality intervals (CLIs) consist of a number of cycle points corresponding to the end of each cycle containing the pattern of references. This number is known as the rank. The

chances of finding substructure within a BLI are related to its rank, number of references in the BLI, the size of the activity set, and the lifetime of the interval in references. Substructures are BLIs contained by BLIs, and their usefulness is dependent upon how much they are referenced. By focusing on the detection of locality intervals and how much useful substructure they contained, a structure parameter was developed to help determine the degree of substructure within a CLI. For a structure parameter equal to 1, the hierarchy of locality intervals is at its lowest level. Conversely, a large structure parameter indicates there is much useful substructure. Using reference strings from some Algol-60 programs they checked their hypothesis concerning the structure parameter. An equation is developed for the cost associated with this event using the average real time required to transfer a segment from secondary memory to primary memory. An expression is then given for the critical value of the structure parameter, since average access time is dependent on the system and this plays a part in determining the cost of descending in the hierarchy. They assert that a measure such as the structure parameter can be used to help control the window size of a working-set algorithm.

Building on previous work by Koh, who used the CLI model to allow for an adaptive-window size using the working-set, Hartley looks at some modified versions of the adaptive-window policy [Har88]. Some of the problems that are considered are the ratio of the memory-access times in the hierarchy and CLIs containing CLIs. Estimated values of the structure parameter are used to determine how the window size should grow. The problem is whether or not structures within CLIs should be traversed. The adaptive-window policy is compared to the working-set policy in terms of memory allocation and space-time execution for various memory-access times. In all cases, the adaptive-window policy performed as well or better than the working-set.

Lenfant looks at the BLI model of program behavior in comparison to the working-set model [Len76]. Using the LRU stack model of program behavior, he focuses on the size of working-sets, the probability that the top i pages in the LRU stack are an activity set, the lifetime of BLIs,

and the probability that a working-set is in fact an activity set. The two models are compared using the results of two programs. He finds that the window size of a working-set has an effect on the probability that it is an activity set and is dependent upon the program being considered. He also discovers that the working-set model and BLI model are not equivalent since the working-set is often not equivalent to the activity set. He concludes the concept of the BLI is important for studying program behavior but states its implementation in a memory management scheme is too complicated.

Masuda investigates a way to detect program localities using the source code of a program and a reference string of the program in execution [Mas83]. The model he presents relies on loop structures since this is where, according to him, the property of locality in a reference string is most likely generated. The source program is used to build a flow graph which is based on three primary structures: the process block, the decision block, and the repetitive-contour block. The repetitive-contour block is important to this model since this block contains the loop structures in which statements are executed more than once. Portions of memory that are referenced are divided into segments. The segments which are referenced by the program are denoted on the flow graph which can now be called a segment flow graph. By using the contour blocks and the segments they contain, locality sets and their associated lifetimes found in the reference string can be identified. These locality contours (LCs) constitute the phases in a manner similar to the BLI model. The LC model is compared to the BLI model for a given program. By visual inspection, it is shown that the LC model is similar to the BLI model in its identification of phases. Both demonstrate the hierarchical nature of phases, however, the LC model is able to eliminate some of the levels contained in the BLI model which contribute little to the characterization of the program's behavior. Multiple levels BLIs from the BLI model seem to characterize the same behavior whereas each level of the LC model characterizes different aspects of the program's behavior.

2.5 Other Program Behavior Studies

Ferrari proposes programs which are better suited to the strategies of a storage system can be obtained by tailoring programs to the models of program behavior upon which these strategies are built [Fer75]. The main objective of restructuring is to enhance a program's 'spatial locality' by making "spatially continuous those parts which are likely to be referenced in temporal proximity [Fer75]." Ferrari reviews the working-set, LRU, First-in First-out (FIFO) and the models which they exploit and introduces a five-phase process to restructure a program. Tailoring algorithms which follow these five phases are discussed and their performance is compared to another tailoring algorithm known as the Critical Working Set (CWS) and to no tailoring algorithm at all. The results indicate the restructuring algorithms provide better performance by keeping the number of missing pages and number of excess pages to a minimum.

Bunt and Murphy introduce a measure of program locality using a hyperbolic distribution known as the Bradford-Zipf distribution in [BM84, BMM84]. Their measure of locality is given as $L = B\bar{n}$, where B is a term derived from the Bradford-Zipf distribution and $\bar{n}$ is a term defined as the average item productivity. The former term is used to characterize the concentration aspect of locality, while the latter is used to characterize the persistence aspect. The authors show restructuring does indeed result in improved locality. It can also be used to generate synthetic reference strings, to identify and analyze phase behavior, and to predict paging performance.

They further expand Madison and Batson's idea of BLIs and characterize program behavior by dividing it into phases and transitions [MB88]. They found phases are longer than transitions and have many more references. They also provide data to support the separate treatment of instruction and data references because instruction references have a stronger tendency to re-reference the same page.

McNiven and Davidson look at the memory referencing behavior of programs to reduce memory traffic. Using a technique known as trace flattening, which reduces the effects of poor compila-

tion and architecture dependence, a trace is broken up into values [MD88]. A value is considered to be the single use of a memory location. Values are grouped into classes. Bounds on the classes can be calculated to minimize the traffic associated with a given class of values. Traffic is divided into two parts: initial and excess. Initial traffic is the traffic generated from the reading of data and instructions for the first time. Excess traffic is the traffic generated by reading into memory data and instructions which were previously in memory. McNiven and Davidson find that the inter-reference time and lifetime of a value is related to the reduction of traffic and that 'dead' values and long inter-reference times are a detriment to the reduction of traffic. They propose the compiler can determine which values are dead and what the inter-reference times will be and that this information can be imparted to a program-controlled cache to enhance performance and reduce traffic.

A Markov model of program behavior built using a program's trace data is presented and reviewed by Bogott and Franklin [BF75]. A synthetic trace is generated from the Markov model and is compared to the trace from which it was derived. Simulations using the traces are run and fault probabilities are plotted against corresponding memory sizes using three different replacement policies: FIFO, LRU, and Random. Their results indicate that for most memory sizes the synthetic trace overestimates the actual fault probabilities. This is due to the Markov model's inability to define specific access patterns by allowing paths to exist which may not have existed in the original trace.

Voldman and others present a novel approach to program behavior and its effect on caches by looking at the fractal dimension of cache misses to characterize an executing program [VMH+83]. This fractal approach makes use of hyperbolic distributions which relate the fractal dimension to the notion of bursts during which cache misses occur and gaps which are the periods between bursts. Bursts are also hierarchical in that bursts are made up of smaller bursts. By using three different

workloads from differing environments, the fractal dimension is measured for each workload. The results agree with expectations of the complexity of various environments.

2.6 Analytical Cache Models

Although trace-driven simulation has provided benefits in the advancement of cache design, it lacks the ability to provide a better understanding of how a cache should be designed to exploit the property of locality. Agarwal, Horowitz, and Hennessy have developed an analytical cache model in an effort to do this [AHH89]. Their model (hereafter referred to as the AHH model) uses values which are obtained from address traces used in trace-driven simulation. The AHH model uses different cache parameters such as cache size, block size, and degree of associativity to predict performance. Agarwal and others define four categories which cause cache misses: start-up effects, non-stationary behavior, intrinsic interference, and extrinsic interference. Start-up effects account for the misses that occur as the working-set of references is loaded into the cache. The miss ratio during this period is also known as the cold-start miss ratio. Non-stationary behavior accounts for the change in the working-set as the program progresses through execution. Intrinsic interference considers the finite size of the cache and the fact that blocks in the program may conflict with each other causing one another to be removed from the cache. Extrinsic interference takes into account the misses that result from a multiprogramming environment. Using a fixed block size, expressions are developed in the AHH model to account for each category's contribution to the miss rate.

The effects of block size are then incorporated into the AHH model by looking at the distribution of run lengths from a trace. A run can be defined as the maximum stream of references to sequential memory locations [AHH89:195]. Increasing block size takes advantage of spatial locality in the program. In their model of spatial locality, the n-stage Markov model (n being the length of the longest run) shown in Figure 2.2 is reduced to two states. This reduction in the number of states is due to the fact that references fall into one of two categories: those that are part of a

run and those that are not. The AHH model is further refined to account for extrinsic interference and an expression which accounts for its contribution to the miss rate is developed. The AHH model is then compared with trace-driven simulations for caches with varying parameters using a random replacement policy. The results of this comparison indicate the analytical model predicts a lower miss rate than what is actually measured during simulation. Two reasons given for the lower rate predicted by the analytical model are that program blocks are not uniformly distributed (an assumption that was made earlier) and that the set of possible conflicting blocks is larger than what was assumed [AHH89:206].

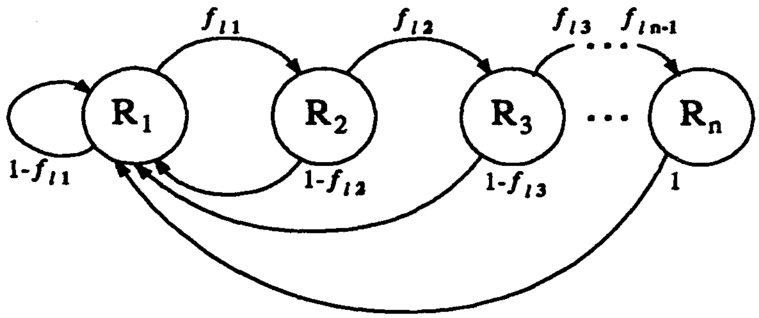

Figure 2.2. AHH model of runlengths (spatial locality).

Another analytical model, a stochastic model for the generation of memory references by a program, is presented in [FR90] and refined in [FR91]. Fricker and Robert derive explicit expressions for the main characteristics of these sequences of references to include working-set behavior and cache miss rates. The model represents locality and the randomness in which the references occur. Their mathematical model considers four parameters: cache size, block size, associativity, and the locality of the program. The model is then used to derive the working-sets and is compared with the IRM. The working-set in this sense has been defined to be the set of addresses referenced.

Analytical expressions for miss ratios are developed. They state that although access times of the various levels have been neglected in their model as well as many others, the influence of access times has a strong influence in the actual cache design.

Singh, Stone, and Thiebaut also develop an analytical model for fully associative caches by starting off with an expression for the temporal behavior of unique references of a program and then modifying this expression to account for block sizes. Their model for the number of unique references uses the power law and is given in the expression $u(t, L) = WL^a t^b d^{\log L \log t}$. L denotes the block size and t denotes time. W, a, b, and d are all constants which are parameters of the model. W is related to the initial working-set size of the trace, a is related to the locality of references in the trace, and b and d are also related to the locality of references in the trace and are also used in deriving a [SST88:3,5]. An important aspect of this model is the reliance on what appears to be a logarithmic relationship between $u(t, L)$ and L (varying with a constant t). There is a slight curvature in the lines that is not modeled. Singh and others develop an expression to find the miss ratio of a cache with a given block size. Another important aspect that is modeled is the logarithmic relationship between the cache size and the miss ratio. Singh and others also discuss how they derived workload parameters from the single trace that was used.

Using data from Smith's 1987 article concerning line-size choices for caches [Smi87], they take data from the design-target miss ratios and plot them with respect to cache size instead of block size. They found straight lines on log/log plots indicating a logarithmic relationship. It was noted that Smith did some smoothing of the data. Singh and others compare their model to the AHH model and concede the AHH model has a wider range of application since it covers set-associative caches and small caches. Small caches are not modeled well by the AHH model since small caches are unable to hold the initial working-set of a program. However, the AHH model is more complex than their model since there are more parameters in the AHH model. The straight-line phenomena on the log/log plots are not specifically modeled by the AHH model, yet they are present. When

the start-up and non-stationary misses are removed from the AHH model, the remaining misses of the two models resemble each other. These remaining misses were plotted by Agarwal on a log/log scale as a function of cache size and a straight line was given for the miss rates [SST88:11]. This was predicted by their model.

2.7 Structural Locality

The spatial and temporal aspects of programs are well known and have been exploited through the use of various prefetching strategies and replacement policies. Structural locality is less well known, and structural aspects typically have not been exploited. The term, "structural locality," appears to have originated at the University of Illinois where they attempted to make use of this locality in a computer design known as the Structured Memory Access (SMA) architecture. The organization of this computer was designed specifically "to take explicit advantage of a program's structure and of the regular patterns in which data structures are referenced [PD83]." Their endeavor in building a computer to overcome the von Neumann bottleneck did not take hold; however, work by Sohi, Thazhuthaveetil, Pleszkun and others have tried to take advantage of this structural locality.

Current work by Sohi also involves taking advantage of the structural nature of programs by using compiler techniques and a programmable cache to decrease the average memory-access time of memory [SH90]. Structural locality of reference was exploited by Thazhuthaveetil and Pleszkun during their study of LISP programs [TP87]. They define structural locality of reference as the property of a data accessing stream that "describes the extent to which the different elements of a complex data structure are referenced in that stream close together in time." To detect structural locality, they used traces of LISP list accesses that were annotated with primitive list-manipulating functions and used the list references to generate list sets. List sets are defined as a set of structurally related list references. Structurally related implies that the lists are either identical or they are the

CAR [2] or CDR [3] of one another. The sets were then analyzed for structural locality by looking at the lifetime of the list sets as well as the percentage of total references contained in the list sets. The lifetime of a list set was defined as the temporal distance between the first and last members of the list set. It was found that a small number of list sets with long lifetimes comprised most of the references in the traces. The authors state these list sets "should constitute locales of high structural locality of reference in the list structure." They also found that these list sets also exhibited a high degree of temporal locality as well. They conclude that spatial and temporal localities detected in other studies were probably a result of the regularity in referencing behavior or structural locality. These studies applied to LISP machines as well as general purpose machines.

Hobart used structural locality to help characterize and model the behavior of symbolic workloads in a LISP environment [Hob89]. His model of program behavior exploited structural locality by using the LRU stack distance string to see what differences existed between the memory-accesses of symbolic programs versus those with more conventional numerically intensive programs. The structural locality he found to exist in the symbolic workloads may have applications to other programming languages, workloads, and architectures. Occurrences of consecutive same stack distances were used to indicate the presence of structural locality since references would occur in the same order in which they were previously referenced. The two-state Markov model he developed is shown in Figure 2.3. One state corresponds to new references being pushed onto the stack and the other to old references being re-referenced. Transitions between the two states take into account the different sequences of references which are possible. The two transitions from the old state back to the old state differentiate between old references with the same stack distance (SSD) and references with a different stack distance (NSSD). The probability of making the SSD transition turned out to be higher for the symbolic workloads than conventional workloads. For all workloads, the instruction references had a high probability of SSD of 0.816.

[2] The CAR function, which stands for Contents of the Address Register, returns the first element of a list

[3] The CDR function, which stands for Contents of the Destination Register, returns a list with all but the first element

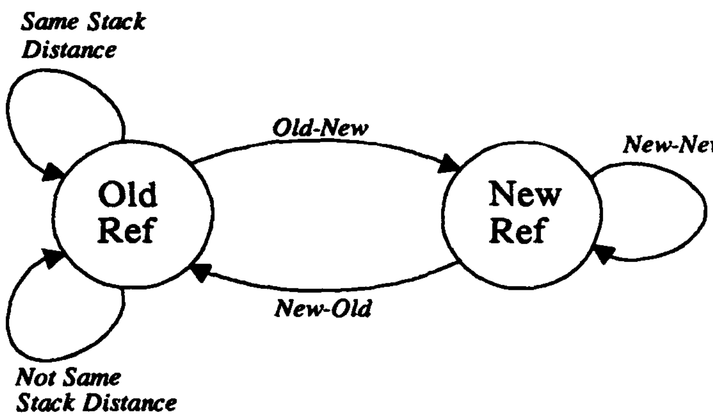

Figure 2.3. Hobart's Memory Reference Behavior (MRB) Model

Bletzinger further refined Hobart's two-state Markov model [Ble92] by expanding the SSD transition into several states corresponding to the length of the same stack distance runs as shown in Figure 2.4. This diagram uses SSDL to denote a same stack distance run of length L. The termination of an SSD run is denoted by the transition NSSDLO and NSSDLN when runs are terminated by an old and new reference respectively. Bletzinger's model also accounts for runs of new references. NNL is used to denote a new reference run of length L. New runs can only be terminated by an old reference (NNLO). In his study of structural locality, he found the distribution of runs lengths of same stack distances to be dependent upon program design.

2.8 Tracing Techniques

2.8.1 ATUM Traces. There does not appear to be a set of standard traces which represent a variety of workloads and computing environments. Perhaps the most widely used traces are the ATUM traces produced by Agarwal, Sites, and Horowitz. ATUM stands for Address Tracing Using Microcode. Their method of obtaining trace data modifies the microcode of the VAX 8200 to record the memory references of the CPU [ASH86]. In every instruction-execution microroutine

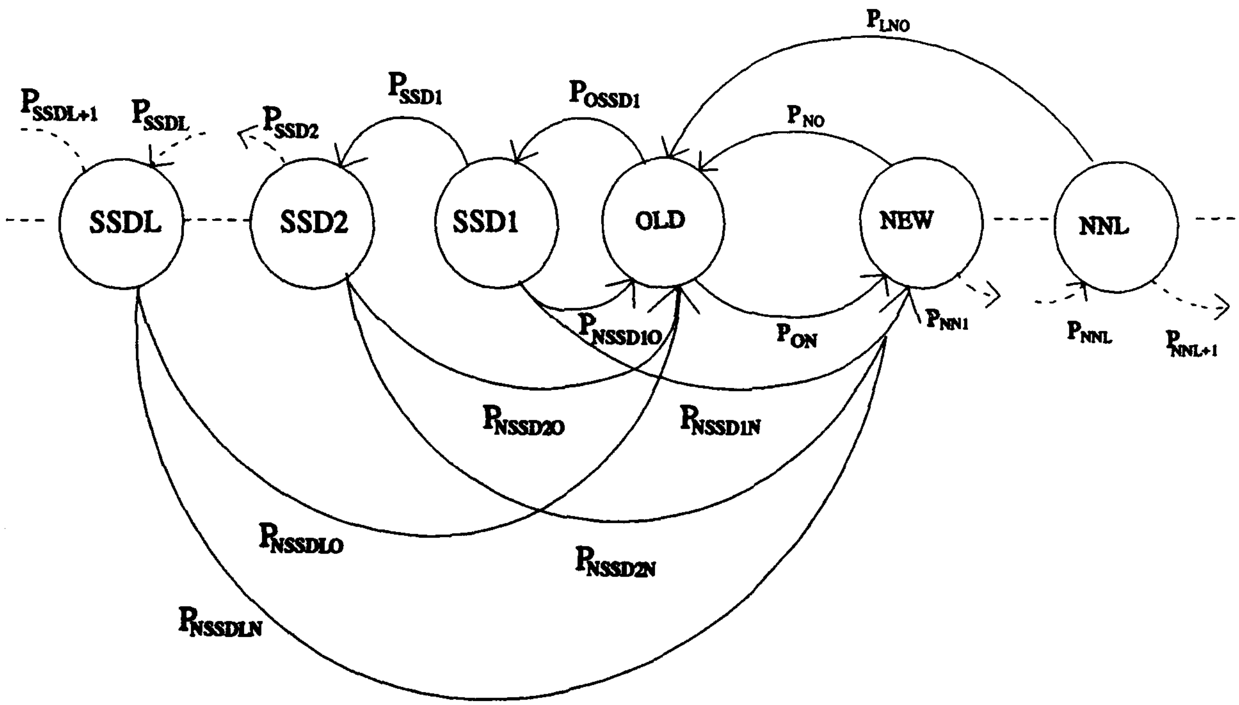

Figure 2.4. Bletzinger's Modified Memory Reference Behavior (MRB) Model

that referenced memory, the microcode was modified to record the referenced memory address. However, not all references were able to be recorded since memory limitations prevented all the necessary microcode modifications. This technique does allow references from process switching and system calls to be recorded. For this research, two different sets of ATUM traces were available. The first set of ATUM traces were obtained through the distribution of Mark Hill's Dinero cache simulator. To work with the simulator, these traces were at the byte level. The second set of ATUM traces were obtained from Agarwal at the Massachusetts Institute of Technology (MIT). These traces, which have been preprocessed, are at the word level and have been used in much of Agarwal's published research. The Dinero set of ATUM traces will be referred to as DIN while the MIT ATUM traces will be referred to as MIT. Three traces are common to both the DIN and MIT sets. A discussion of the differences between the DIN and MIT traces is located at Appendix B. For reasons discussed in Chapter 3 and Appendix B, the MIT traces have been favored over the DIN traces. A description of the ATUM traces used in this research can be found in Table 2.1.

2.8.2 LISP Traces. A slightly different technique was used by Hobart to record the memory references of LISP programs executing on the TI Explorer II [Hob89]. Instead of modifying the microcode routines as Agarwal and others did, he modified the page fault handler routine to record memory addresses. Since the page map table was also modified to cause the page fault handler to be executed on every memory reference, he was able to obtain the necessary trace data. Descriptions concerning the LISP traces used in this research can also be found in Table 2.1.

2.8.3 DLX Traces. Another group of traces were obtained from a simulation of the DLX microprocessor presented in Hennessy and Patterson's textbook on computer architectures. The DLX is a 'mythical' RISC (Reduced Instruction Set Computer) processor which contains elements of both the MIPS (Microprocessor without Interlocking Pipe Stages) and SPARC (Scalable Processor Architecture) microprocessor architectures. A description of the DLX can be found in [HP90]. Three programs written in C were compiled for the DLX, 'executed', and traced using the simulator.

Table 2.1. Trace Descriptions

TRACE NAME	SOURCE	DESCRIPTION
biaslisp	TI Exp	circuit analysis
boyer	TI Exp	theorem prover
compile-rb	TI Exp	Lisp compiler - Phase 1
compile-str	TI Exp	Lisp compiler - Phase 2
fft	TI Exp	numeric computation of Fast Fourier Transform
glisp-comp	TI Exp	expert system tool - compilation of GLISP expert system
glisp-pay	TI Exp	expert system tool - execution of GLISP expert system
qsim	TI Exp	qualitative reasoning
reducer	TI Exp	symbolic computation
tmycin	TI Exp	expert system tool
dec0.000	DIN	DECSIM (beh simulator) simulating cache H/W
fora.000	DIN	FORTRAN compile
forf.003	DIN	FORTRAN compile
fsxzz.000	DIN	file check program
ivex.000	DIN	Interconnect Verify checking net lists in VLSI chip
linp.000	DIN	Linpack benchmark
lisp.000	DIN	Lisp runs of BOYER (a theorem prover)
macr.000	DIN	macro assembler
memxx.000	DIN	memory checker
pasc.000	DIN	Pascal compile of microcode parser program
savec.003	DIN	samples of C compiler
spic.000	DIN	SPICE simulating 2-inp tri-state NAND buffer
ue02.000	DIN	UNIX emulator
dec0.001	MIT	DECSIM (beh simulator) simulating cache H/W
dec1.001	MIT	DECSIM (beh simulator) simulating cache H/W
dia0	MIT	diagnostics program for the VAX
forl.000	MIT	Fortran compile of Linpack
forl.001	MIT	Fortran compile of Linpack
ivex.000 (dup)	MIT	Interconnect Verify checking net lists in VLSI chip
ivex.003	MIT	Interconnect Verify checking net lists in VLSI chip
lisp.000 (dup)	MIT	Lisp runs of BOYER (a theorem prover)
lisp.001	MIT	Lisp runs of BOYER (a theorem prover)
pasc.001	MIT	Pascal compile of microcode parser program
spic.000 (dup)	MIT	SPICE simulating 2-inp tri-state NAND buffer
spic.001	MIT	SPICE simulating 2-inp tri-state NAND buffer
umil1	MIT	MIPS instruction level simulator
umil2	MIT	MIPS instruction level simulator running TLB
cc1	DLX	GCC using own source files as input
spice	DLX	SPICE circuit simulation
tex	DLX	Common TeX within man pages as input

These traces, like the Dinero ATUM traces described previously, were to be used with the Dinero cache simulator and are at the byte level. The descriptions of these programs can also be found in Table 2.1.

2.9 Summary

This chapter has provided the background upon which this research is based. Terms that are used in this thesis have been defined and previous approaches to the modeling of program behavior have been reviewed. The LRU stack model has shown the most promise in these approaches.

Analytical models have been developed as an alternative and an aid to trace-driven simulation in order to provide better insight into behavior. By incorporating structural locality into a model of program behavior, behavior can be understood by the localities which a program displays. The programs used in this research were obtained using various tracing techniques. They provide the data upon which a unified model is based.

III. Methodology

3.1 Overview

This chapter discusses the methodology used in extending the MRB discussed in the previous chapter to account for spatial and temporal locality. The chapter begins with a discussion of various measurements taken on the traces to further characterize them. Measures established by Hobart in [Hob89] are extended to other traces. Data concerning spatial, temporal, and structural locality are provided and discussed. From this, a model which accounts for the spatial, temporal, and structural aspects of a program is developed and explained.

3.2 Trace Characteristics

Tables A.1 and A.2 list the traces and various data about them. The LISP traces all consist of 450,000 references. The ATUM traces typically have between 250,000 and 450,000 references while the DLX traces run much longer with 800,000 to 1,000,000 references. The number of instruction, data, data read, and data write references are given along with the percentage of instruction references. The percentage of data reads to the total number of data references is also given. Note that the instruction references for the LISP workloads average approximately 33.3%, while the two workloads which Hobart called conventional (or numeric), Biaslisp and FFT, had a much smaller percentage of instruction references, 12.7%. Without these two workloads present the instruction average would be about 38.5%. The MIT ATUM workloads had about a 50.0% instruction reference average while the DIN ATUM traces had about a 53.1% average. The DLX workloads had a high percentage (75.3%) of instruction references due to the RISC processor from which the trace is obtained. RISC processors are characterized by simple instructions and fixed instruction lengths. Often several instructions are required to accomplish a task that a CISC processor accomplishes with a single instruction.

Because of the two different levels of traces, word and byte, the characteristics described later cannot always be used to compare word-level traces to byte-level traces and must be taken in context with traces of the same level. Due to previous research and reasons discussed in Appendix B the word level traces are of primary interest and will be the only traces used in Chapter 4.

The percentage of unique references to the number of dynamic references is provided in Table A.2 as an indication of the amount of new referencing activity contained in the trace. In some of the LISP, DIN, and MIT traces the number of unique instruction and data references does not equal the total number of unique references contained in the trace. This is understandable where data reads and data writes make up the data references since many of the data references are both read from and written to; however, when the same address is used for both an instruction fetch and a data reference, this is more puzzling. Having duplicate instruction references and data references indicate that the code is self-modifying. One possible explanation for this are procedure calls where the location of a procedure is stored and a jump is made indirectly to the stored location. Situations such as this should be avoided if split instruction and data caches (Harvard architecture) are to be implemented. The DLX traces did not have this problem.

In looking at the traces, the frequencies of an address occurring were also noted. It has been stated that 90% of the references can be accounted for with 10% of the code [HP90]. Figures 3.1-3.4 show what these figures were for the the overall traces for the LISP and MIT ATUM workloads. Each workload set was divided into two plots for clarity and the grouping is alphabetical. Although the percentages vary for the various traces this rule generally applies with 20% of the unique references accounting for 80% of the total references encompassing a wider range of the traces. This indicates that much of the time is spent re-referencing locations already referenced and that the most frequently referenced locations are the ones which should be kept in the fastest memory. In general, it appeared that the MIT ATUM traces adhered to the 90/10 rule better than the LISP traces.

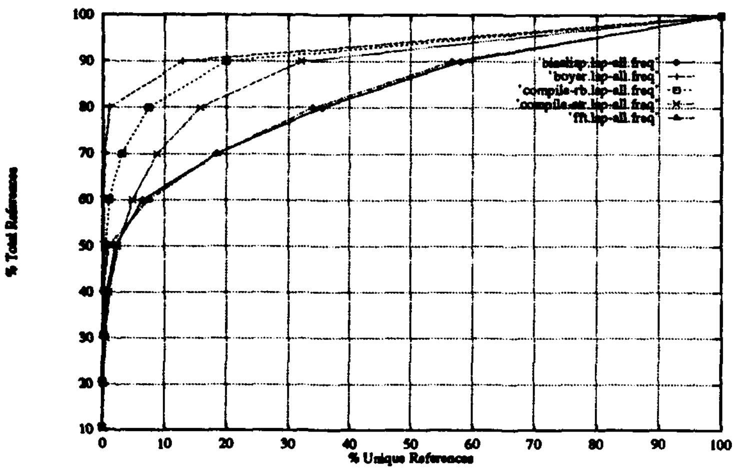

Figure 3.1. Lisp Traces (plot 1) - ALL Refs Frequencies

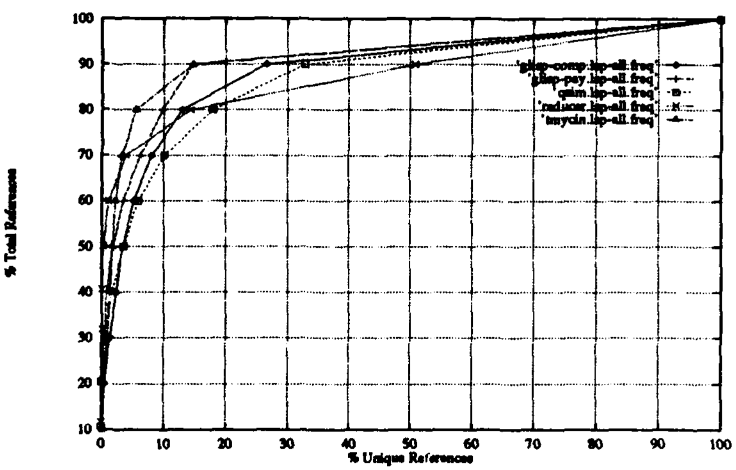

Figure 3.2. Lisp Traces (plot 2) - ALL Refs Frequencies

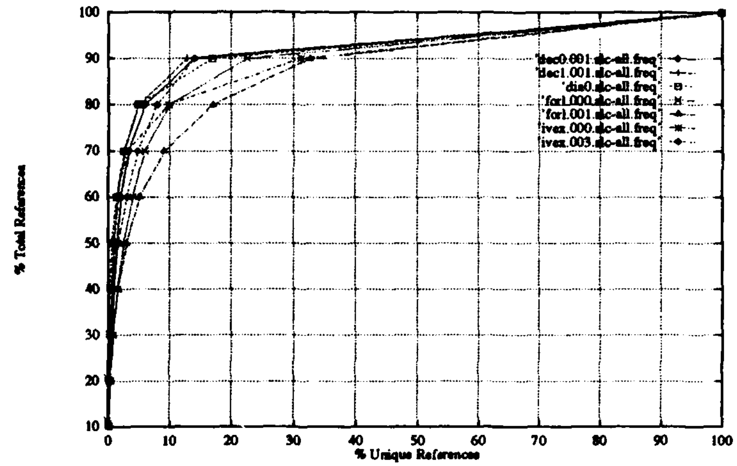

Figure 3.3. ATUM MIT Traces (plot1) - ALL Refs Frequencies

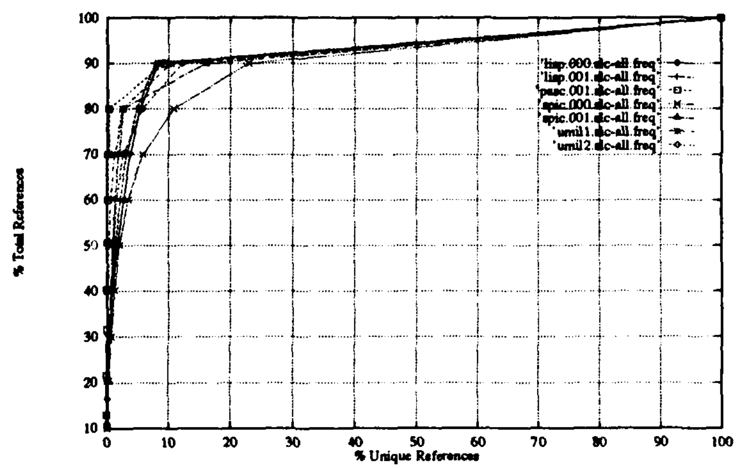

Figure 3.4. ATUM MIT Traces (plot2) - ALL Refs Frequencies

For all of the traces, plots were made of the address references, their corresponding LRU stack distance string, the growth of the LRU stack, the cumulative histogram of LRU stack distances, the individual histogram of spatial distances, and the cumulative histogram of spatial distances. Spatial distance is defined to be one plus the number of address locations located between two succeeding references and is found by subtracting the address of the previous reference from the address of the next reference. Each of these plots were made for the five types of reference traces: all, instruction, data, data read, and data write. These plots provide familiarity and insight into temporal and spatial aspects of the traces.

Address reference plots were made to see the virtual address space as it was being referenced. For the LISP traces the address space was 25 bits wide for a possible 2^{25} address locations. Memory addresses were referenced in the range between 0 and 9898680h (1.0 x 10^7). The heap appeared to be around 1FF2B60h (3.35 x 10^7) in those traces in which this area was referenced extensively. For the MIT traces, 2^{30} word locations exist in the virtual address space. The major area of referencing was in the low space near 0h, the space around 1F972880h (5.3 x 10^8), and the upper space of 3FC6E780h (1.07 x 10^9). These plots show where in the address space the references were and when they were taking place. Figures 3.5-3.8 are examples of these plots. These plots support Stone's notion of how the references are distributed [Sto87:27]. He states that there are a few regions which have a high probability of being referenced and other regions where this probability is low or moderate. The rest of the address space has a very low probability of being referenced. These references are not uniformly distributed but instead are concentrated in certain areas of the address space for various periods of time.

The LRU stack distance string plotted over time showed the temporal characteristics of the reference trace. This plot showed how many unique references were made before a reference was referenced again. New references were indicated by using a stack distance of zero. This plot provided insight into how often and how deep the LRU stack was referenced. A recurring pattern

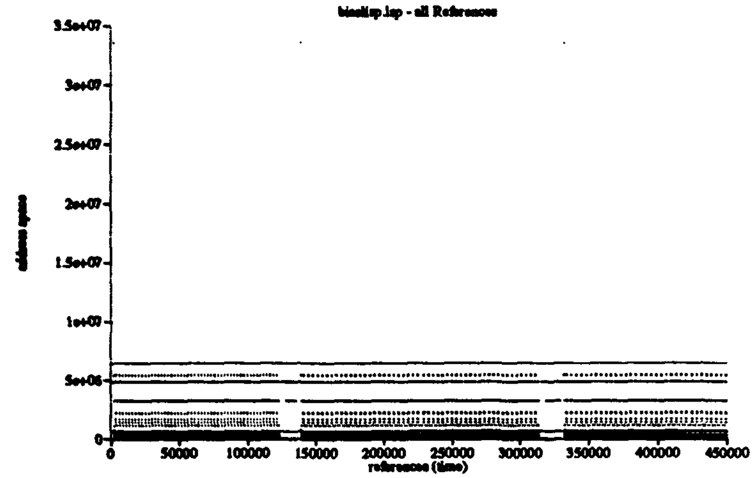

Figure 3.5. BIASLISP - ALL Refs Address Reference Plot

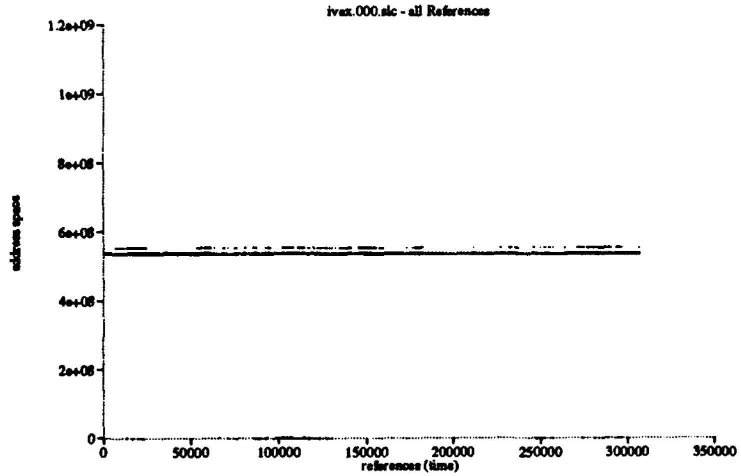

Figure 3.7. IVEX.000 (MIT) - ALL Refs Address Reference Plot

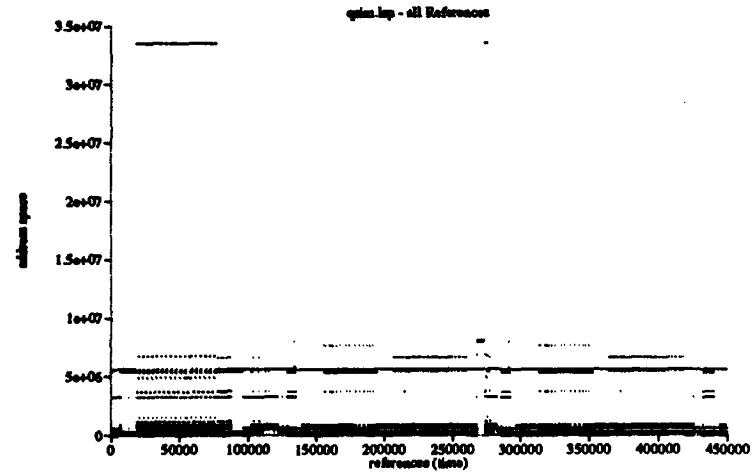

Figure 3.6. QSIM - ALL Refs Address Reference Plot

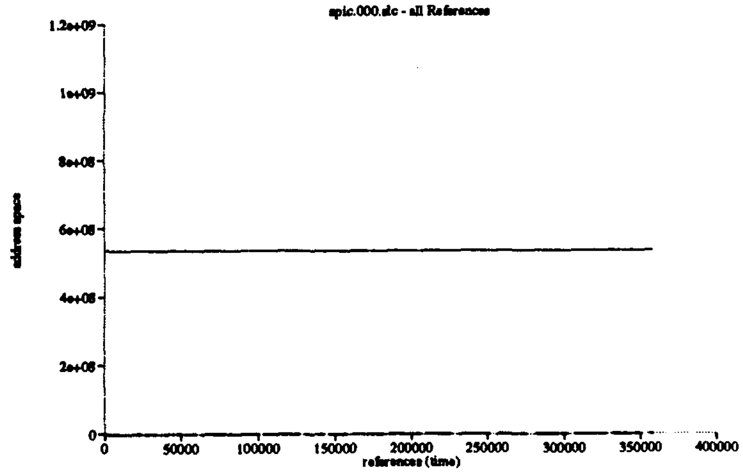

Figure 3.8. SPIC.000 (MIT) - ALL Refs Address Reference Plot

of stack distances would occur in some of the plots of various traces indicating that memory was being referenced in a manner similar to what it had been before. This should not be confused with the same-stack-distance (SSD) metric which has been used as a measure of structural locality. This metric would be evident in these plots by horizontal lines indicating that the same stack distance was being referenced over time. Examples of these plots are shown in Figures 3.9-3.12. In Hobart's research he noticed that the two conventional LISP workloads (Biaslisp and FFT) had a high percentage of references deep into the stack by looking at the LRU stack distance cumulative histogram. We thought that these references deep into the stack might indicate a return to an earlier processing phase. However, the LRU stack distance plot shows that these deep stack references occur uniformly throughout much of the trace's run.

The stack growth plot provides an overview of how the generation of new references would occur since each new reference increases the stack size by one. Figures 3.13-3.16 are examples of these plots. A slope of zero on this plot indicates periods in which no new references occur. Changes in this slope indicate a change in locality and in program behavior. Whereas the percentage of unique references gives only an indication of the extent of new referencing behavior, the stack growth plot shows how and when these new references are distributed. The instruction plots tended to have more periods with a zero slope than the other types of reference plots. Overlaying this plot with the LRU stack distance plot shows how the stack is being utilized. Memory addresses which have not been referenced for the longest time define the bottom of the stack.

The cumulative histogram of LRU stack distances provide a measure of the stack size necessary to account for and capture any number of references. New references with stack distances of zero are included in this histogram. This plot is the same as that shown by Wong and Morris in determining the LRU cache hit function [WM88]. These plots take into account the unique references and give an indication of how big the LRU stack must be in order to contain a given percentage of the references. For the two LISP workloads discussed earlier these plots had a sharp rise to account

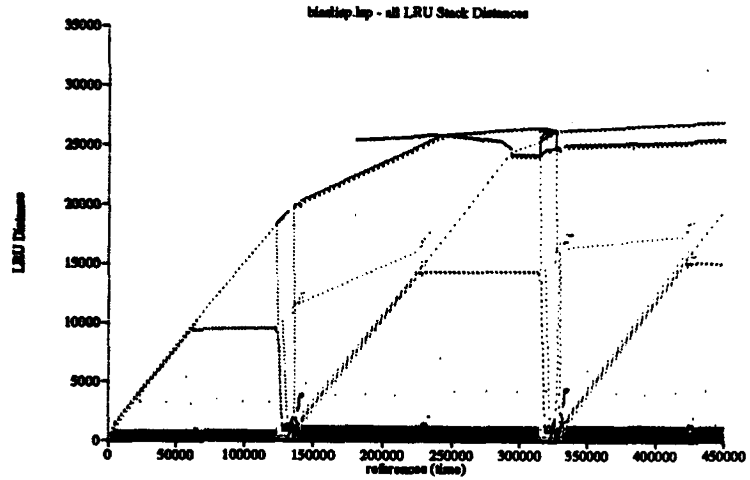

Figure 3.9. BIASLISP - ALL Refs LRU Stack Distances

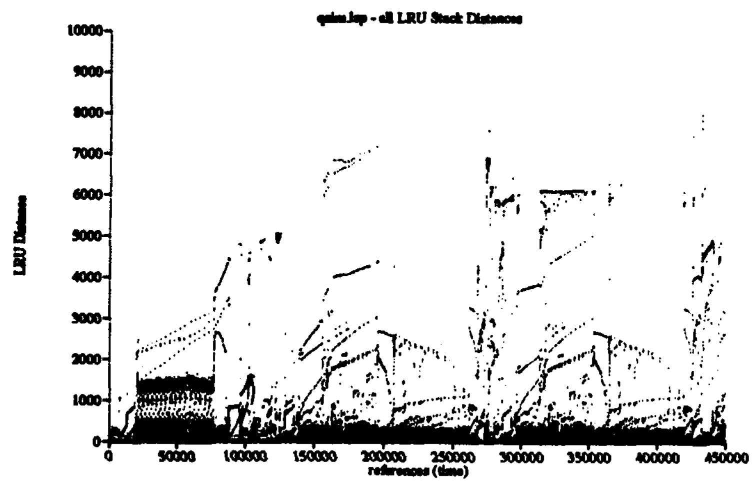

Figure 3.10. QSIM - ALL Refs LRU Stack Distances

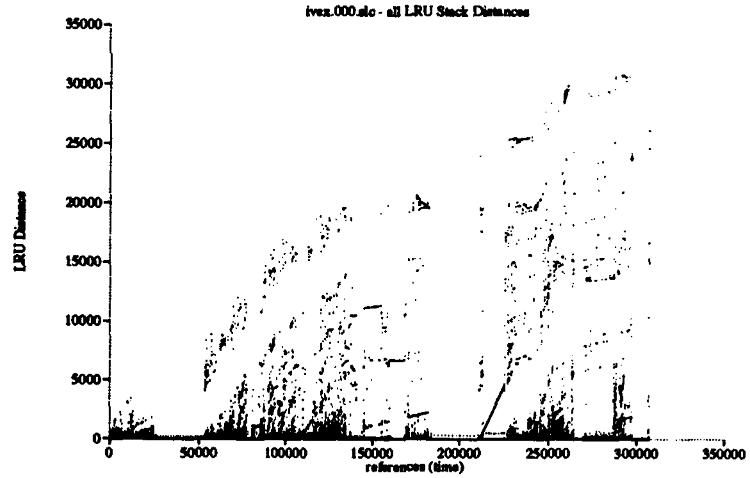

Figure 3.11. IVEX.000 (MIT) - ALL Refs LRU Stack Distances

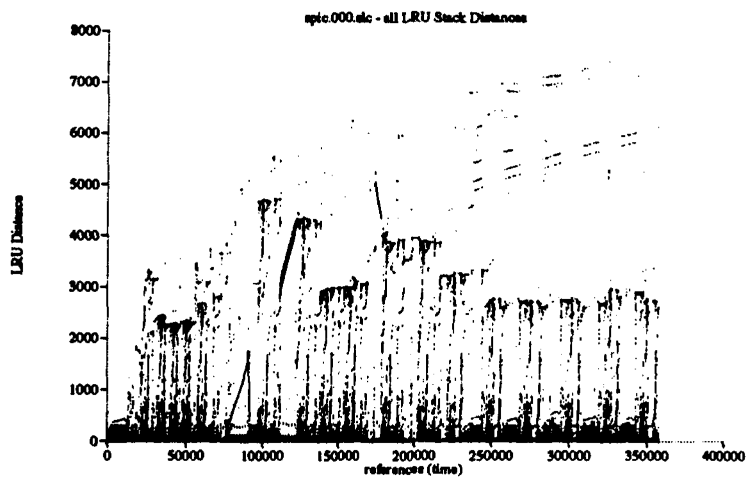

Figure 3.12. SPIC.000 (MIT) - ALL Refs LRU Stack Distances

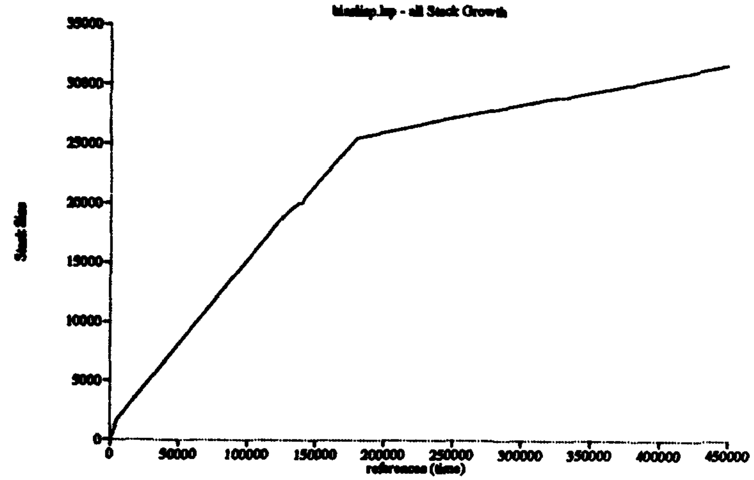

Figure 3.13. BIASLISP - ALL Refs Stack Growth Plots

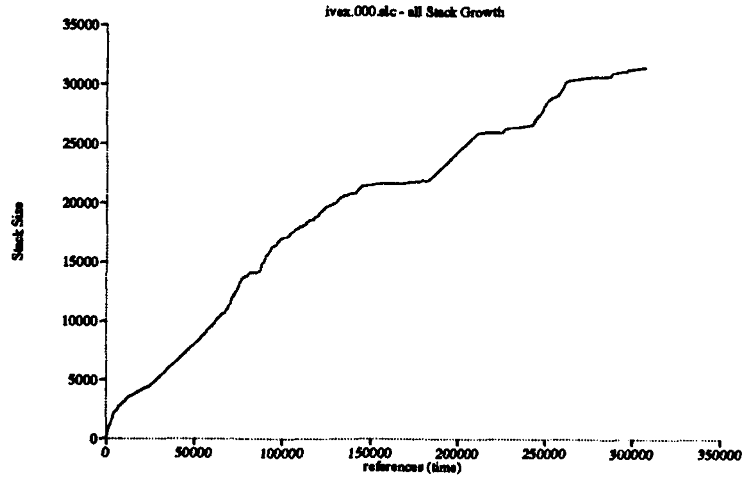

Figure 3.15. IVEX.000 (MIT) - ALL Refs Stack Growth Plots

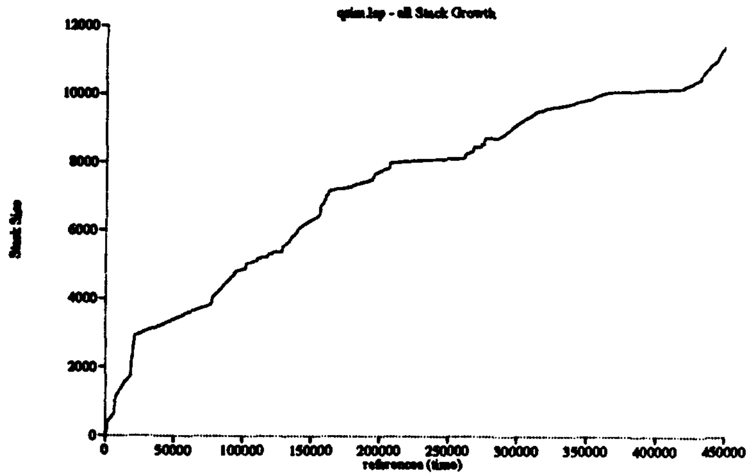

Figure 3.14. QSIM - ALL Refs Stack Growth Plots

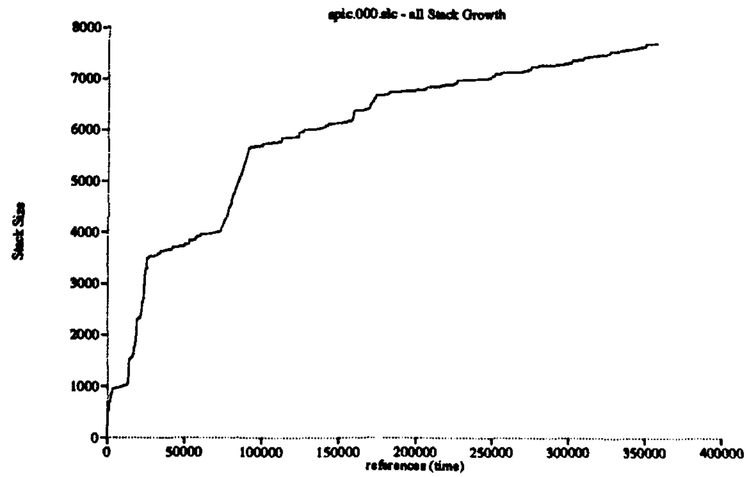

Figure 3.16. SPIC.000 (MIT) - ALL Refs Stack Growth Plots

for a good portion of the references. BIASLISP is among the traces shown in Figures 3.17-3.20 as examples of this plot.

The cumulative histogram of spatial distances is the same as that defined and described by Hobart [Hob89]. Due to the wide range of the spatial distances, the x-axis is the $\log_2$ of the absolute value of the spatial distance. Positive values indicate references in the forward direction of the address space while negative values are used to indicate references in the reverse direction. Hobart noticed that in the LISP traces, spatial distances between references were either within 32 addresses of the reference or if not, then at least 32K addresses from the reference. His plots of the cumulative histogram of spatial distances reveal this. Plots for the ATUM and DLX traces also exhibit these same characteristics. Figures 3.21-3.24 are examples of traces from the different sets. These examples are characteristic of the other traces within the set. Note the zero slope on both the reverse and forward sides of the plot which indicate the low number of references between the 32 and 32K word region. There are some variations in the traces where this region may be a little wider or narrower. The only real exception to this is in the DLX TEXtrace where 10% of the references have a spatial distance between 32 and 32K words.

3.3 Spatial Locality

Hobart's spatial window probability [Hob89], P_{SW}, indicating that the next reference is within 32 addresses is given in Tables A.3-A.7. This measurement was taken using all the transitions in the five different kinds of traces as well as old-new and new-new transitions. $P_{SW_{all}}$ was quite high for the instruction traces with an average of 0.902 for the LISP and MIT ATUM traces combined. For all the types of reference traces, $P_{SW_{new-new}}$ is generally higher than $P_{SW_{old-new}}$. This is understandable since new references would be expected to be located close to one another; whereas, in going from an old reference to a new reference, this is not as likely to happen. The instruction reference traces showed that $P_{SW_{new-new}}$ was quite high (around 0.9) as well, indicating

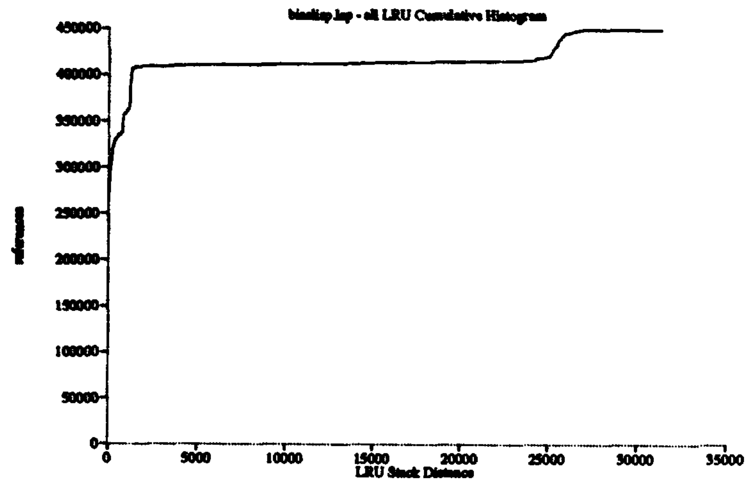

Figure 3.17. BIASLISP - ALL Refs LRU Cumulative Histogram

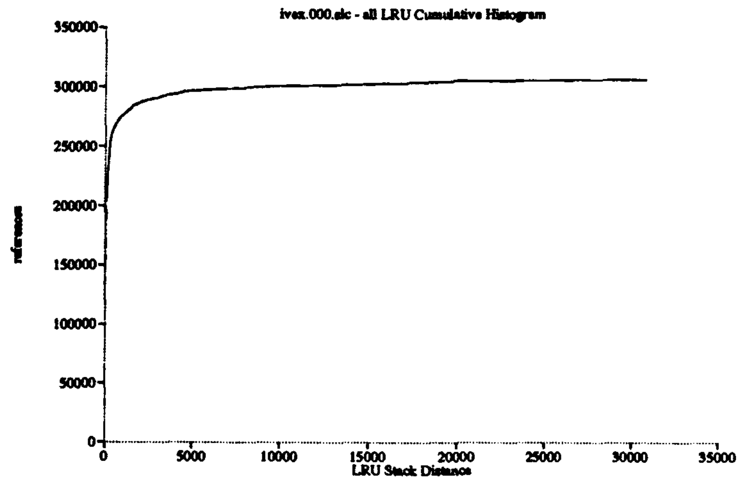

Figure 3.19. IVEX.000 (MIT) - ALL Refs LRU Cumulative Histogram

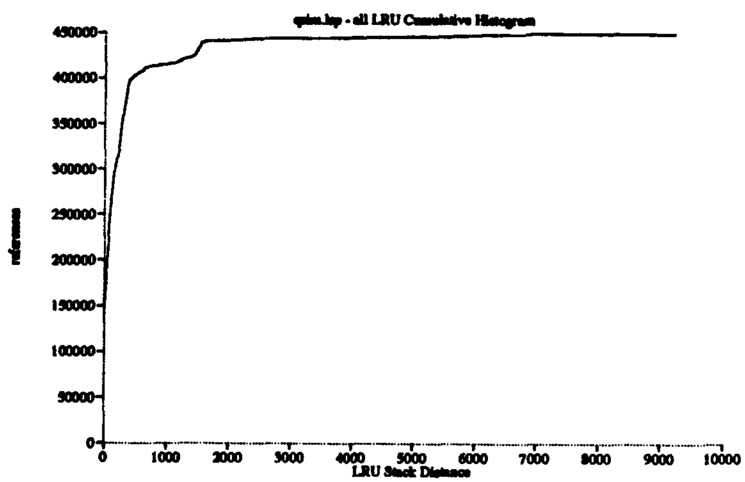

Figure 3.18. QSIM - ALL Refs LRU Cumulative Histogram

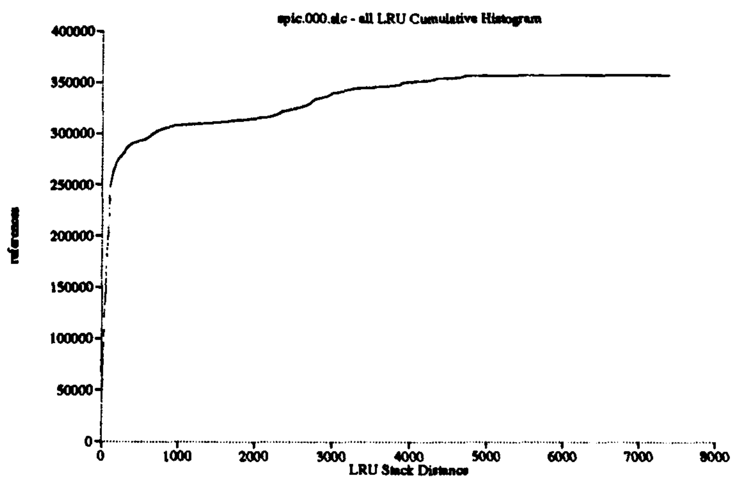

Figure 3.20. SPIC.000 (MIT) - ALL Refs LRU Cumulative Histogram

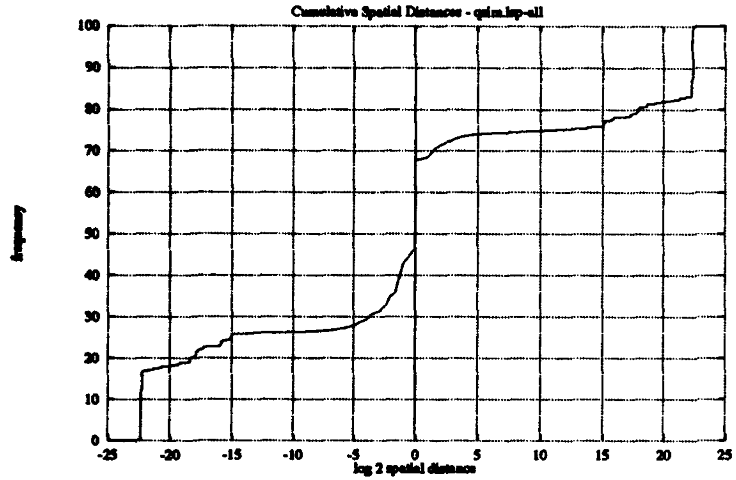

Figure 3.21. QSIM All - Spatial Cumulative Histogram

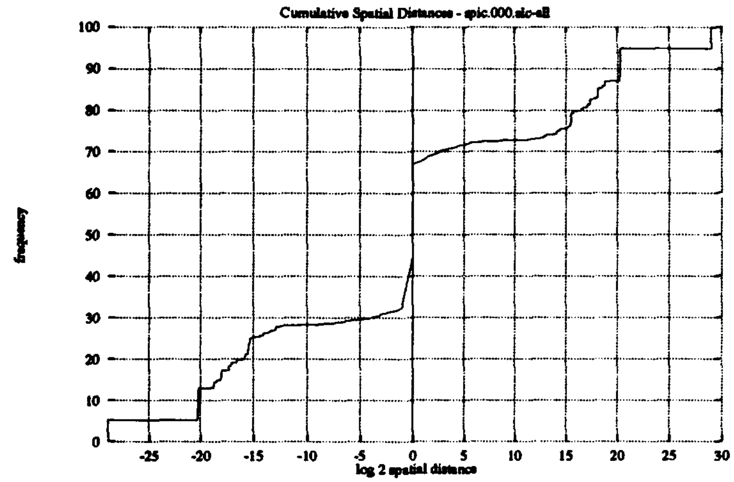

Figure 3.23. SPIC (MIT) All - Spatial Cumulative Histogram

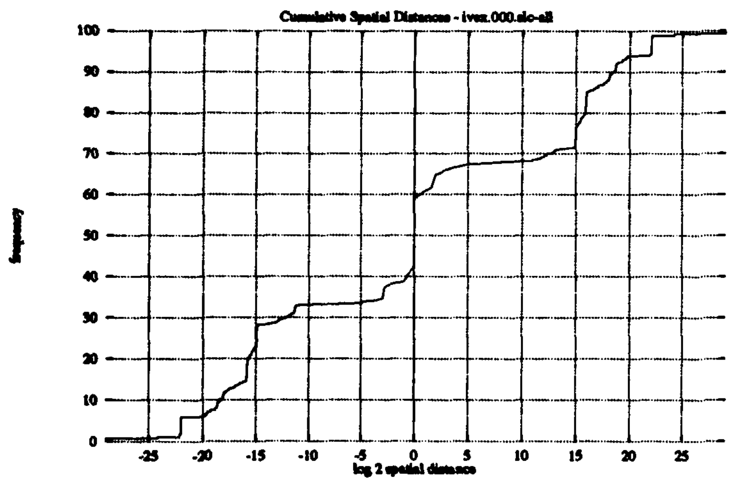

Figure 3.22. IVEX (MIT) All - Spatial Cumulative Histogram

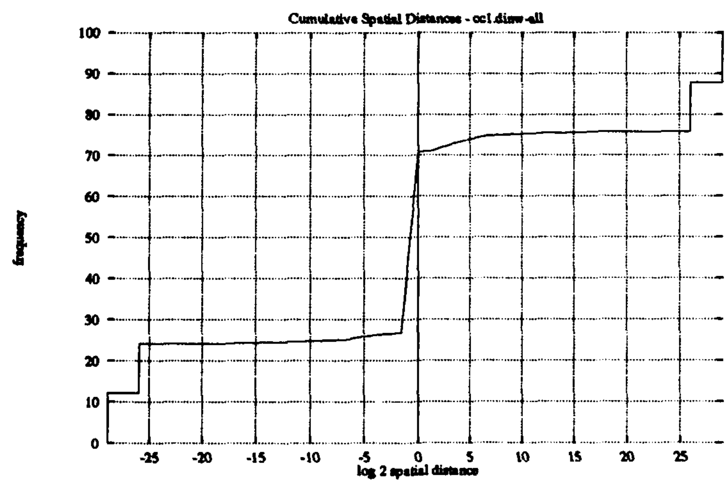

Figure 3.24. CC1 All - Spatial Cumulative Histogram

the strong presence of spatial locality. In the other types of reference traces, $P_{SW_{new-new}}$, is above 0.5 varying with the type of trace. The one exception to this is the DLX TEXtrace in which the data references have a very low value. Separating the data references into reads and writes, the value for $P_{SW_{new-new}}$ is almost 1 and indicates the data is being read in one location and then being written to another in a sequential fashion. The write reference traces had values of $P_{SW_{new-new}}$ which averaged around 0.888 for the LISP and MIT ATUM traces combined.

In addition to P_{SW}, the spatial distances between new references were recorded and binned. This is similar to the data presented in the cumulative spatial histogram plots, however, these measurements measure the spatial distances between the new references. This includes not only the distance from new reference to new reference, but also the spatial distance from the previous new reference, going through a sequence of old references and encountering the next new reference. Old-New as used in this measure has a different connotation than that in the P_{SW} measure. References to the next sequential location were fairly common occurrences and are noted in Tables C.1-C.15 as S Jmp. Differences between the word-level and byte-level traces are apparent when looking at this measure as the word-level traces had a greater tendency to reference the next word address, while the byte-level trace did not have as great a tendency to reference the next byte address. Spatial distances of 32 addresses in the forward direction were considered small jumps (SmJmp) while spatial distances in the opposite direction were considered small backward jumps (SBJmp). At the other extreme, spatial distances greater than 32K addresses in the forward direction were considered big jumps (BgJmp) while spatial distances in the other direction were considered big backward jumps (BBJmp). Spatial distances which fall between these two extremes were considered medium jumps (MdJmp) and medium backward jumps (MBJmp). This data was collected for new-new transitions (same as P_{SW}), old-new transitions (spatial distance is between previous new reference and next new reference), and for both of these transitions combined (spatial distances between all new references). Spatial distances larger than 32 addresses are generally

symmetrical indicating that a transition from one locality to another would eventually result in a transition back to the previous locality.

The next consideration in the measurement of spatial locality was to find out how much of a spatial prefetch block is being used. The unique references were extracted from the original trace and then changed into blocks by dividing the addresses by the block size. References to words within the blocks were recorded in order to compute the resulting average number of unused references in the blocks of the trace. Unused references the first time a block is referenced were also computed by counting only the consecutive references to a block after its first reference. These measurements showed that the benefits of prefetching the entire block of references were attained not only during the first time the block was required, but also during subsequent references to the block. This is indicated in Tables D.1-D.15 by the decrease in non-referenced words between the first time that a block was referenced and its resulting references.

3.4 Temporal Locality

Using measures established by Hobart concerning stack distance thresholds, the percentage and size of the stack required to capture 90%, 95%, and 99% of the references which are re-referenced is provided in Tables A.8-A.12. The other number in the tables is the actual percentage of references which crosses the threshold. These tables are a numeric characterization of the LRU cumulative histograms described earlier; however, unlike the LRU cumulative histograms, these tables do not include new references as part of the stack. The higher the percentage of references accounted for (the number in the first column), the more temporal locality exists in the trace. The two conventional LISP workloads discussed earlier also exhibit a high degree of temporal locality, particularly with the data reference traces.

For example, the LISP BIASLISP trace has 4.7% of the unique words referenced accounting for 90% of all the references which are referenced again. The stack or memory size required to

capture this 90% is 1,491 words. In order to capture 95% of all references, the stack size would grow to 25,283 words which are 79.6% of all unique references. The 99% threshold requires a stack size of 26,075 words accounting for approximately 82% of the unique references. The other 18% of unique references are either not referenced soon enough to be contained in the stack or are never referenced again.

These measures were also applied to the blocked versions of the traces for block sizes of 4, 8, and 16 words. Blocking exploits the spatial locality contained in the traces. The results from these measurements indicate that the 'blocking' of the references initially increases temporal locality. By converting the size of the stacks using blocks into words, the P_{LRU}s and the stack size to obtain them can be compared. In order to capture 50% of the references, it was found that larger block sizes are able to capture more references using a smaller stack. However, as the threshold increases to the 95% and 99% levels the blocking may become less advantageous in capturing the temporal locality. In order to capture all the references which are referenced again, individual word references result in a smaller stack size than the spatial prefetch blocks. The size of the stack required to capture various levels of temporal locality was often dependent upon the trace. For fully associative caches which use LRU replacement, the P_{LRU} can be used to accurately determine what the hit rate will be for any given cache size.

The MIT ATUM trace DEC0.001 illustrates this. To capture 50% of all references, the regular word-level trace requires a stack size of 61 words accounting for 1% of all unique words referenced. Using a block size of 4 words, a stack size of 12 words is required to capture over 50% of all the references. For 8 and 16 word blocks, this 50% threshold requires the stack size to be 16 and 32 words respectively. With all three block sizes, the percentage of unique blocks is less than 0.2%. In order to guarantee that 70% of the references are captured, the word level trace needs a stack size of 127 words while the blocked traces need a stack size 40 words for the 4 and 8 words blocks and 48 words for the 16 word block. As the threshold increases, the size of the 4, 8, and 16 word block

stacks becomes a larger percentage of original word trace stack size. At the 99% threshold the stack sizes required for these 4, 8, and 16 word blocked traces are greater than what is demanded by the original word-level trace by using 3,148, 3,416, and 3,472 words respectively. The word-level trace only requires 3,010 words to capture the same percentage of old references.

The two conventional (numerical) LISP traces discussed earlier, depict the effect of blocking on temporal locality. Figures 3.25-3.26 show the 'jump' discussed earlier and the effect of blocking on the temporal distance string. Distances of the blocked LRU Distance strings have been multiplied by the number of words in the block in order to compare the strings. This measurement provides some information concerning the relationship between spatial and temporal locality. The tables of P_{LRU} in Appendix D indicate that exploiting spatial locality helps to enhance the temporal locality and capture a majority of the references. The original P_{LRU} for the word thresholds are given followed by P_{LRU} thresholds for block sizes of 4, 8, and 16 words. The thresholds account for 50, 70, 90, 95, 99, and 100 percent of the references. The advantages of exploiting the spatial locality diminish for temporal locality as all the references become accounted for.

3.5 Structural Locality

Structural locality was investigated using the metrics established by Hobart which looked at same stack distance transitions. Hobart defined P_{SSD} as:

P_{SSD}: The probability that the subsequent reference will have the same stack distance as the previous reference given that the previous reference was an old reference [Hob89].

Using Hobart's model (see Figure 2.3), the following equations are given for each of the transitions in the model.

$$P_{SSD} = \frac{N_{SSD}}{N_{SSD} + N_{NSSD} + N_{ON}} \tag{3.1}$$

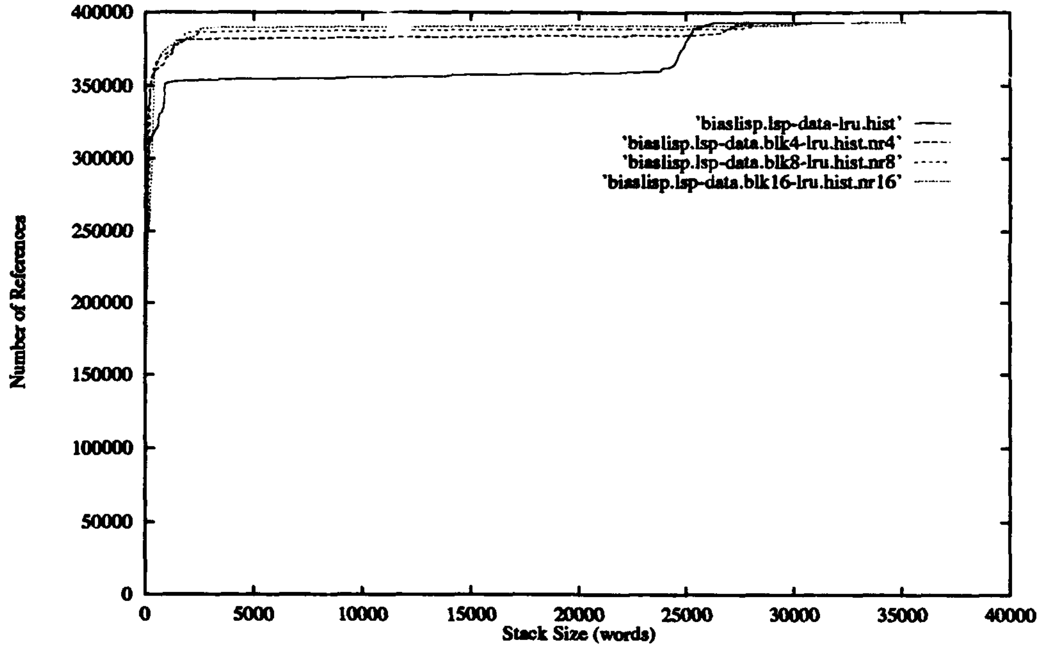

Figure 3.25. LRU Cumulative Histograms for Biaslisp - Data.

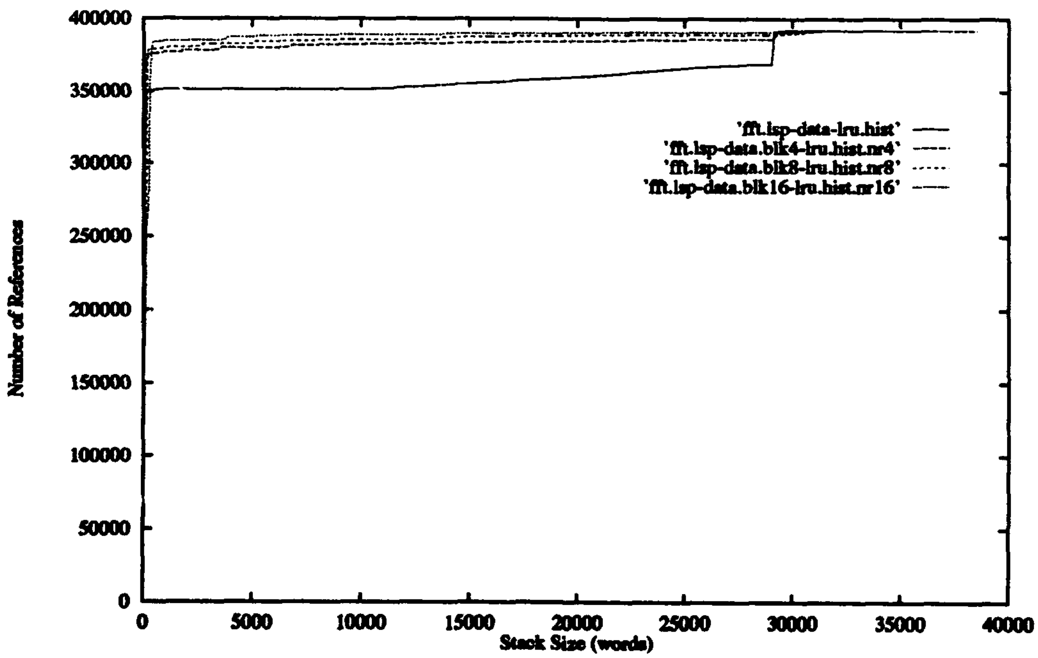

Figure 3.26. LRU Cumulative Histograms for FFT - Data.

$$P_{NSSD} = \frac{N_{NSSD}}{N_{SSD} + N_{NSSD} + N_{ON}} \tag{3.2}$$

$$P_{ON} = \frac{N_{ON}}{N_{SSD} + N_{NSSD} + N_{ON}} \tag{3.3}$$

$$P_{NO} = \frac{N_{NO}}{N_{NO} + N_{NN}} \tag{3.4}$$

$$P_{NN} = \frac{N_{NN}}{N_{NO} + N_{NN}} \tag{3.5}$$

where

N_{SSD} is the number of Same Stack Distance transitions.

N_{NSSD} is the number of Old-Old transitions minus N_{SSD}.

N_{ON} is the number of Old-New transitions.

N_{NO} is the number of New-Old transitions.

N_{NN} is the number of New-New transitions.

These state transition probabilities for the traces are listed in Tables A.13-A.17. It is worth noting that P_{SSD} is quite high in not only the instruction references of the LISP traces, but also in the instruction references in the ATUM and DLX traces as well.

The state transition probabilities were also recorded for blocks of 4, 8, and 16 words for the word-level traces and are located in Tables F.1-F.15. This provides insight into the relationship between spatial and structural locality. The important thing to notice here is the resulting decrease in P_{SSD} from the original trace. Subsequent increases in block size raise the probability of same stack distance but this is due to increases in runs of stack distances equal to 1 (SD=1) which indi-

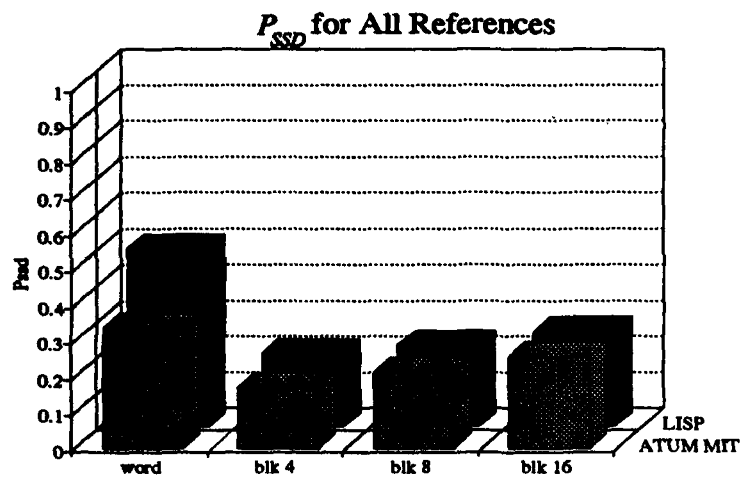

Figure 3.27. P_{SSD} for All References.

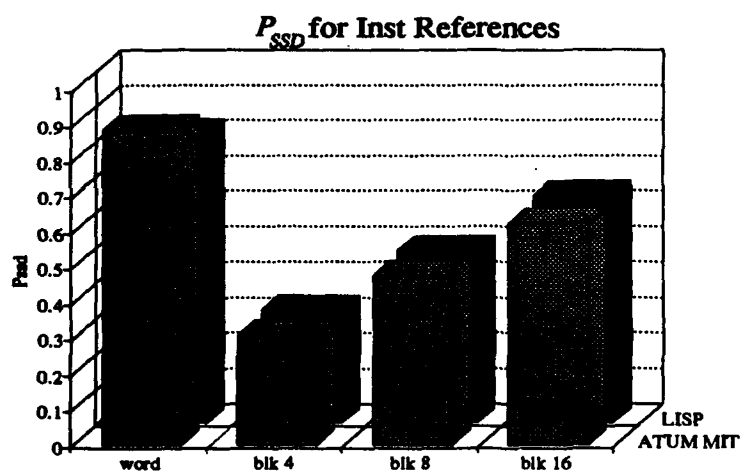

Figure 3.28. P_{SSD} for Inst References.

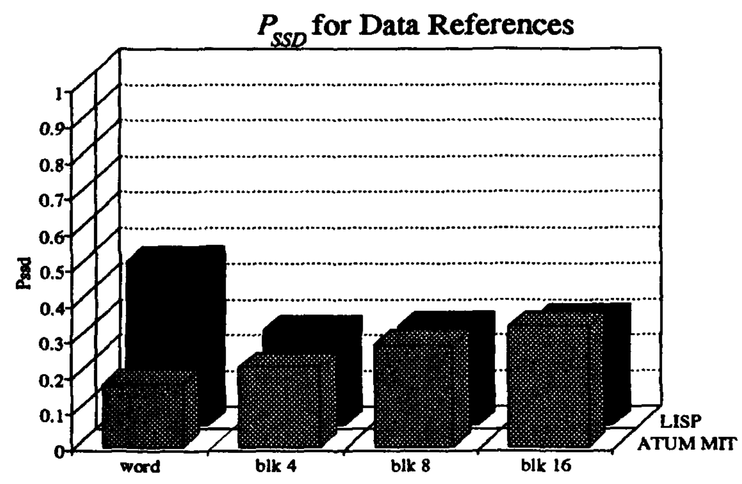

Figure 3.29. P_{SSD} for Data References.

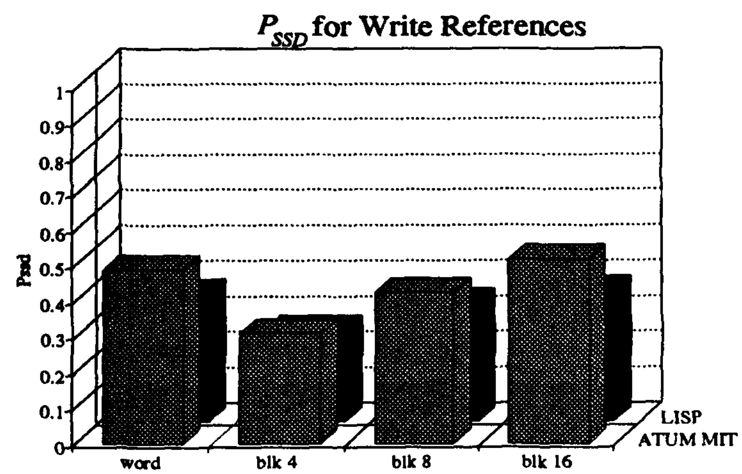

Figure 3.30. P_{SSD} for Write References.

cates the re-referencing of the block just referenced. Structural locality which had been measured by using P_{SSD} is now hidden by the spatial locality of the blocks. Figures 3.27-3.30 show this relationship of how P_{SSD} is initially reduced by going to a small block size and then grows as the block size increases. Due to the similarity in the data and read reference traces, the read reference bar graph is not shown. The ATUM MIT data traces are an exception in that the word-level P_{SSD} is less than the blocked traces.

Figure 3.31 presents the P_{SSD} data across the LISP and ATUM MIT traces for the five different types of reference traces. The LISP bar is on the left while the ATUM MIT bar is on the right. Although the two types of traces are similar in the characteristics for P_{SSD} when blocked, the word-level trace has a higher value for P_{SSD} in the data and read trace types for the LISP workloads.

The increase in runs of stack distances equal to 1 was verified by looking at the run length distributions and their contributions to P_{SSD} and $P_{new-new}$. These distributions were derived from the LRU distance strings of the trace and the corresponding block sizes of 4, 8, and 16 words. As the block size increases, same stack distance run lengths generally become shorter and more frequent. For runs on new references, run lengths are shortened, particularly as the block size becomes larger. This is due in part to the reduced number of new references from converting the trace to block references. There is a significant increase in runs which specifically reference a stack distance of one. This means that a reference has been made to the block which was previously referenced as discussed earlier.

3.6 Model Derivation

In a unified model, each type of locality should be accounted for. The spatial locality measures of non-referenced words (Appendix D) in the blocked traces clearly indicate that spatial prefetching is advantageous since many of the prefetched references are used later. The closer the references

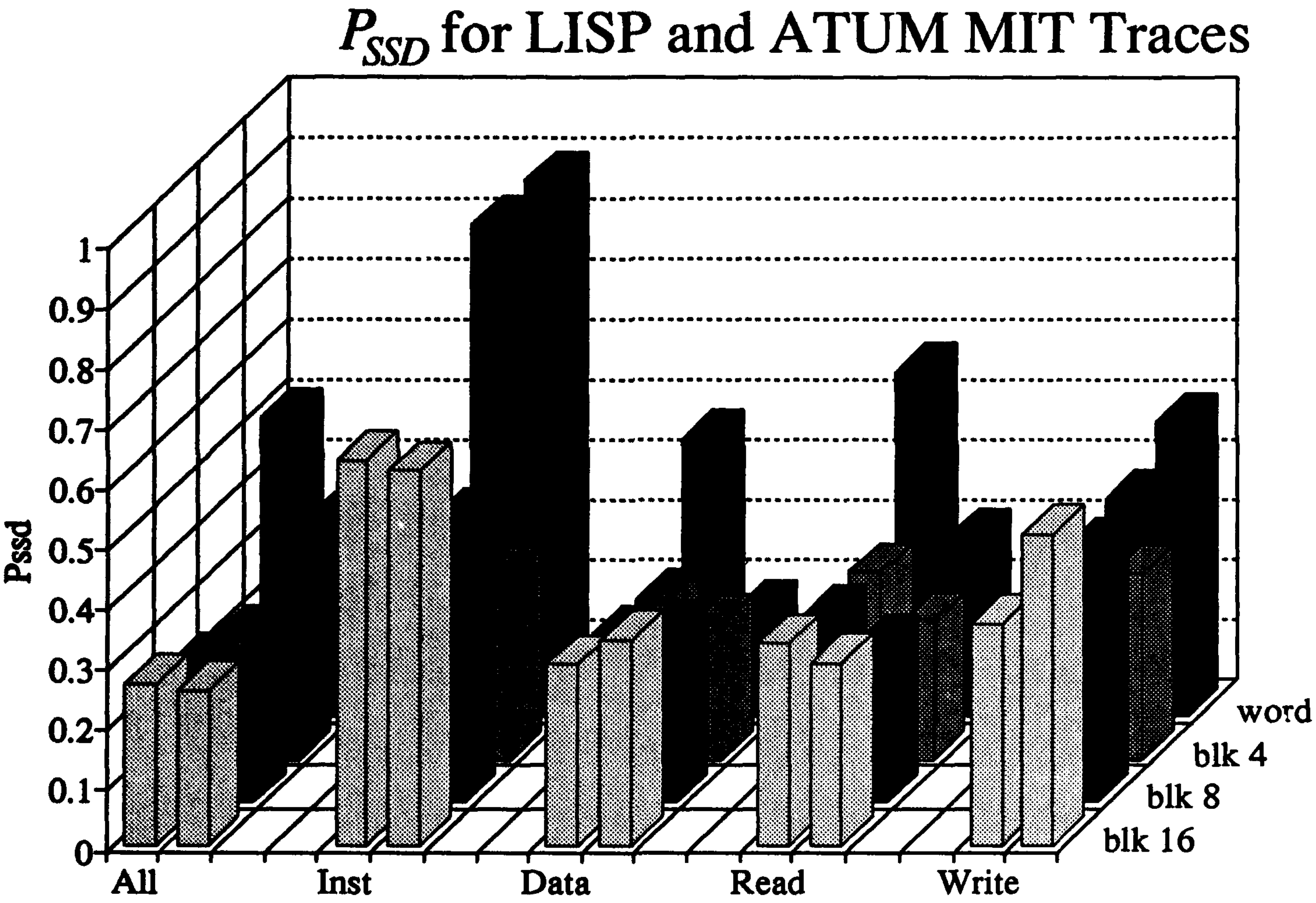

Figure 3.31. P_{SSD} for LISP (left) and ATUM MIT (right) Traces.

are to each other spatially, the greater the benefit from spatial prefetching. The temporal locality measures provide an indication of the stack size required to achieve guaranteed access to a given number of references. In trying to build a model which accounts for the structural locality as well, spatial locality has been measured by the number of different locations which are used within a block. Reuse of previous references would be accounted for by temporal locality.

Dividing a trace's references by the various block sizes resulted in a number of references to the previously referenced block. This effect shows up as a stack distance (SD) of 1 in the LRU distance string. With this information, Hobart's MRB model was modified as shown in Figure 3.32 to account for this behavior. The new state had been expanded to three states to account for the new runs of SD=1 and for new references in blocks which had been previously been referenced for some SD>1. Like the "New" state, runs of SD=1 from the "Old" state were accounted for by an additional state for the old references. All possible transitions which may occur have been included in this model.

Bletzinger's work with structural locality expands the number of states in the model by converting the SSD transition into several states which are dependent upon the distribution of SSD run lengths found within the trace. Although the model is quite complex, an additional state would be useful in determining if further expansion would be worthwhile [Ble92].

Some of the transitions do not seem likely to happen. For example, being in the "SD=0 New" state, it does not seem likely that a transition to the "SD=1 Old" state would occur or that being in any of the old states would result in a transition to the new state SD=1. Measurements of the transition probabilities reveal that these transitions and others do, in fact, have a very low probability. Table 3.1 lists the average transition probabilities for each type of reference trace from both the LISP and ATUM MIT traces. It was also found that the probabilities for being in any of the new states is also low. This data justifies reducing the complexity of the model.

Table 3.1. Table of State Transition Probabilities

Transition	LISP all	MIT all	LISP inst	MIT inst	LISP data	MIT data	LISP read	MIT read	LISP write	MIT write
SD=0-SD=0	0.133	0.099	0.137	0.161	0.139	0.125	0.156	0.140	0.021	0.125
SD=0-SD=1(n)	0.462	0.244	0.759	0.745	0.455	0.229	0.253	0.207	0.607	0.486
SD=0-SD>1(n)	0.134	0.136	0.021	0.033	0.108	0.080	0.063	0.079	0.147	0.074
SD=0-SD=1(o)	0.023	0.001	0.003	0.000	0.037	0.023	0.032	0.026	0.006	0.011
SD=0-SD>1(o)	0.248	0.522	0.080	0.060	0.262	0.543	0.497	0.548	0.218	0.304
SD=1(n)-SD=0	0.083	0.152	0.246	0.280	0.077	0.098	0.077	0.122	0.069	0.136
SD=1(n)-SD=1 (n)	0.290	0.213	0.574	0.585	0.291	0.193	0.269	0.263	0.298	0.364
SD=1(n)-SD>1(n)	0.047	0.089	0.041	0.045	0.023	0.052	0.055	0.069	0.039	0.055
SD=1(n)-SD=1(o)	0.042	0.019	0.035	0.018	0.090	0.107	0.090	0.084	0.058	0.097
SD=1(n)-SD>1(o)	0.538	0.526	0.104	0.073	0.518	0.550	0.508	0.462	0.535	0.348
SD>1(n)-SD=0	0.086	0.075	0.163	0.202	0.074	0.050	0.076	0.066	0.026	0.070
SD>1(n)-SD=1(n)	0.385	0.185	0.548	0.560	0.416	0.168	0.151	0.153	0.607	0.408
SD>1(n)-SD>1(n)	0.101	0.103	0.036	0.063	0.057	0.078	0.068	0.057	0.005	0.078
SD>1(n)-SD=1(o)	0.106	0.010	0.122	0.076	0.147	0.037	0.069	0.056	0.022	0.064
SD>1(n)-SD>1(o)	0.322	0.626	0.132	0.099	0.306	0.667	0.636	0.668	0.341	0.381
SD=1(o)-SD=0	0.004	0.005	0.000	0.001	0.004	0.006	0.007	0.007	0.003	0.008
SD=1(o)-SD=1(n)	0.001	0.001	0.001	0.001	0.002	0.002	0.009	0.004	0.001	0.003
SD=1(o)-SD>1(n)	0.015	0.014	0.000	0.001	0.017	0.012	0.014	0.011	0.002	0.013
SD=1(o)-SD=1(o)	0.305	0.387	0.616	0.607	0.313	0.472	0.341	0.416	0.369	0.572
SD=1(o)-SD>1(o)	0.675	0.595	0.038	0.391	0.664	0.507	0.630	0.562	0.625	0.404
SD>1(o)-SD=0	0.006	0.006	0.001	0.001	0.008	0.009	0.010	0.011	0.069	0.015
SD>1(o)-SD=1(n)	0.001	0.001	0.002	0.002	0.001	0.002	0.013	0.003	0.000	0.003
SD>1(o)-SD>1(n)	0.017	0.016	0.001	0.001	0.021	0.026	0.030	0.029	0.204	0.029
SD>1(o)-SD=1(o)	0.311	0.206	0.775	0.761	0.299	0.277	0.236	0.237	0.291	0.506
SD>1(o)-ssd	0.184	0.141	0.110	0.127	0.247	0.135	0.281	0.169	0.124	0.124
SD>1(o)-nssd	0.481	0.630	0.112	0.108	0.424	0.552	0.431	0.551	0.312	0.324

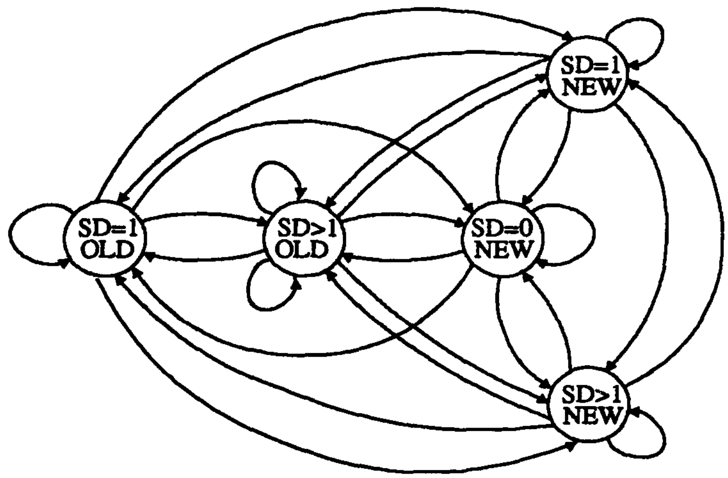

Figure 3.32. Unified Markov model.

In Hobart's original MRB model, the probability of being in the "New" reference state is dependent upon the number of unique references in the trace. Although the two additional new states vary in the probability of being in one of these states, the probability in being in one of the new states is quite low when compared to being in one of the two old states. By collapsing the expanded new reference portion of the model back into a single "New" reference state, this model increases the significance of being in a new reference state while at the same time making the model more manageable. New references can no longer be considered only with a stack distance of 0 in the blocked version of the LRU distance string, therefore the original LRU distance string is used to determine transitions to this state. Figure 3.33 shows this modified version of the unified model. Transitions between the SD=1 state and the "New" state would be expected to have a low probability.

This state also accounts for what Agarwal had described as a run within a block /citeAGARWAL1. Temporal locality can be depicted in this model by using the appropriate probability for being in the LRU stack. Although structural locality is less prevalent due to spatial locality, it is

represented by the SSD transition in the old state. In the next chapter, this model is validated to see how well it depicts program behavior.

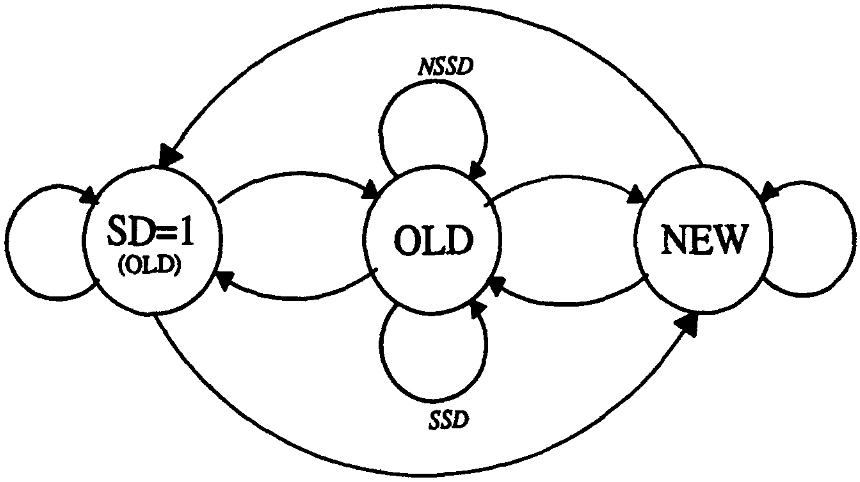

Figure 3.33. Simplified Unified Markov Model.

3.7 Summary

This chapter has discussed various characteristics of the traces with respect to spatial, temporal, and structural locality. Although there are some behaviors which are characteristic to all of the traces (e.g. high degree of spatial and structural locality for the instruction references), the individual traces vary to some degree with one another with some having a higher degree of a certain type of locality than another. The spatial locality measurements indicate that the references contained within a block are not all used when the block is fetched initially, but are likely to be used later during subsequent references. Spatial locality was also used to show that the sequential access of memory, particularly with the instruction traces, is indeed present and can be represented in the LRU stack distance string with a stack distance equal to one. The conflict between spatial locality and structural locality was shown to exist in the blocked version of the traces since the spatial locality has now hidden the structural locality. In the next chapter, an attempt at validating this model is made.

IV. Results

This chapter discusses the results of validating the model introduced in the previous chapter. Due to the varying granularity of the traces, only the LISP and MIT ATUM traces are used since both are at the word level. The model is validated using the entropy measurement of the original string and comparing it to the entropy measurement of a synthetic string generated from the model. The entropy measurement is discussed followed by results in validating the model. Subsequent modifications to the model are made to obtain a 'better' model. This chapter concludes with a discussion of other considerations regarding model validation.

4.1 The Entropy Measurement

Entropy is a measure of the randomness in a sequence of symbols. Randomness can be thought of as information. A sequence of symbols with little randomness contains little information because the sequence is fairly predictable. Therefore, a low entropy measurement is indicative of low randomness and high predictability in the sequence of symbols. Hammerstrom and Davidson used entropy to measure the information content of sequences of memory address references in an attempt to improve the memory/CPU bandwidth and CPU addressing efficiency [HD77].

Shannon showed how entropy can be estimated by finding the occurrences of various sequences of symbols, known as N-grams, in the entire sequence of symbols [Sha48]. In the case of model validation, the various states of the model were used to assign symbols corresponding to the LRU distance string. N-grams correspond to the order of the entropy which is being calculated. In a first order entropy estimation, the entropy calculation is only concerned with each symbol independent of the other symbols. In the second order entropy estimation, sequences of two symbols in succession are considered. In this and subsequent entropy estimations, the N-grams overlap in the sequence of symbols. Shannon gives the equation for estimating entropy as [Sha50]:

$$
\begin{aligned}
F_N &= -\sum_{i,j} p(b_i, j) \log_2 p_{b_i}(j) \\
&= -\sum_{i,j} p(b_i, j) \log_2 p(b_i, j) + \sum_i p(b_i) \log_2 p(b_i)
\end{aligned} \tag{4.1}
$$

N is the order of entropy and the size of the block of symbols.

b_i is a block of $N-1$ symbols.

j is an arbitrary symbol following b_i.

$p(b_i, j)$ is the probability of the occurrence of N-gram b_i, j.

$p(b_i)$ is the probability of the occurrence of N-gram b_i.

$p_{b_i}(j)$ is the conditional probability of the occurrence of symbol j after the block b_i.

Estimating the entropy is accomplished by using the second line in equation 4.1 as it accounts for the smaller $N-1$-grams contribution to the entropy. As N becomes larger, F_N gets closer to the absolute entropy H as seen in equation 4.2 below [Sha50].

$$
H = \lim_{N \to \infty} F_N \tag{4.2}
$$

Because the sequence of symbols corresponding to the trace is finite, the entropy can only be estimated.

4.2 Validating the Model

Validating the model consisted of taking the entropy measurement of the original LRU distance string as it corresponded to the states of the model and comparing this measure with the entropy measurement of a synthetically generated string of symbols. The transition probabilities for the model being validated were taken from the LRU distance string at the same time it was

converted into symbols corresponding to states in the model. An entropy measurement was made of the string of symbols. The transition probabilities were then fed into a synthetic string generator which created a string of symbols corresponding to the model being validated. First, second, and third order entropy calculations were used to compare the synthetic string with the original string.

Due to the inherent nature of Markov models, first and second order entropies of both the original string and the synthetic string should be equal. The first order entropy is a function of the state probabilities which are in turn determined by the transition probabilities. The second order entropy takes into account the previous symbol and the next symbol since a 2 symbol N-gram is being examined. This corresponds directly to the transition probabilities which were used to generate the model. However, in a third order entropy estimation, the Markov model cannot consider a sequence of three symbols unless the model somehow accounts for it. This can be accomplished by limiting transitions in the states of the model so that certain states can only be reached by going through other states first. The goal was to try to get the third order entropy measurement of the synthetic string to be equal to that of the original string.

Each of the LISP and MIT-ATUM traces were measured using block sizes of 4, 8, and 16 for each of the all, instruction, data, data read, and data write traces. Three different synthetic strings were generated for a given block size/type of reference trace with the length of the synthetic string equal to the length of the trace from which it was taken. The first synthetic string used the length of the trace as a seed to the random number generator, while the two other synthetic strings used the time function to generate seeds. Measurements from the synthetic strings were subtracted from the original string to give an indication of the difference between the strings. An average and standard deviation of these differences, as well as, the absolute value was computed.

There are two ways to look at the entropy measure in validating the model: compare the third order entropy of both the original and synthetic string or to have the second and third order entropies of the original string to be equal. The former has been discussed earlier, while the latter

would indicate the Markov model has captured the behavior of the reference string since no further predictability can be made from examining larger sequences of symbols.

In either case, the Markov model would be able to generate a string with the same behavior as the original string.

4.3 Initial Validation Results

With the model shown in Figure 3.33, the first and second order entropies did come out to be very close to equal. Tables G.1 and G.2 show the average differences between the original and synthetic entropies for the strings used in this model. Averages and standard deviations for the first, second, and third order entropies are listed for each of the block sizes corresponding to 4, 8 and 16 words. An overall average and standard deviation is also given. The absolute value of the difference is used for these computations. It was found that the third order entropy would be consistently overestimated using the synthetic string. The third order entropy of the synthetic string was not that different from the second order, indicating that the third order entropy did not show any more predictability in the synthetic string. The third order entropy of the original string, however, was lower than the second order entropy indicating that the original string had more predictable behavior and was less random in its generation of references than the synthetic string.

The various block sizes did not seem to have an effect on the validity of the model (no trends could be observed in the differences in the entropies); however, as block size increased, the estimation of first order entropy also increased. This indicates that the increasing block size reduced the predictability in the LRU distance string provided by the model as state transition probabilities became more uniform. The instruction traces were an exception to this observation. Increasing their block size resulted in a decrease in the first order entropy measurement. Second and third order measurements followed this trend although some traces would go up with block

size of 8 before going down with block size of 16. This tendency to have less entropy with larger block sizes is due to the sequential nature of the memory address references in the instruction traces discussed earlier. In this case, the model, which exploits stack distances equal to one, benefits from increasing block size and makes memory accesses for the instruction traces more predictable.

The entropy measurement also gave an indication of the spatial locality present in terms of stack distance equal to one. For some traces, such as the LISP FFT instruction trace and the MIT-ATUM UMIL data traces, the entropies were quite low in comparison to the other traces of the same type. No noticeable differences were observed between the LISP and MIT-ATUM traces.

4.4 Modified Model Results

In an attempt to bring the third order entropy of the synthetic string closer to the original string, various other models were used to generate synthetic strings and to estimate entropy. The models were of three basic types with varying number of states: ones that exploited the SD=1 transitions (as in the original model), ones that exploited the same stack distance transitions (SSD), and ones that recognized both SD=1 and SSD transitions. Average differences between the first, second, and third order entropies of the original and synthetic strings for each of the various models discussed here are provided in Appendix G.

4.4.1 Four-State and Five-State SD=1 Models The four-state and five-state SD=1 models that were investigated are shown in Figures 4.1 and 4.2 respectively. These models provide additional states to account for SD=1 runs of lengths 1, 2, and 3. Figure 4.1 adds one more SD=1 state to account for SD=1 run lengths of two or greater , while Figure 4.2 adds two more SD=1 states to account for SD=1 runs of three or greater. The longer runs are modeled by staying in the last SD=1 state. As with the original three-state SD=1 model, these two models seem to work best for the instruction traces and worst for the write traces. One exception for the instruction traces are the LISP traces with a block size of four. For these traces the error averages about 0.10 with

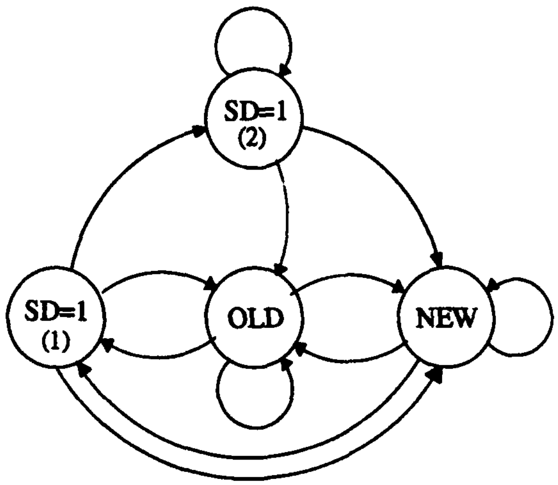

Figure 4.1. Four-State SD=1 Markov Model

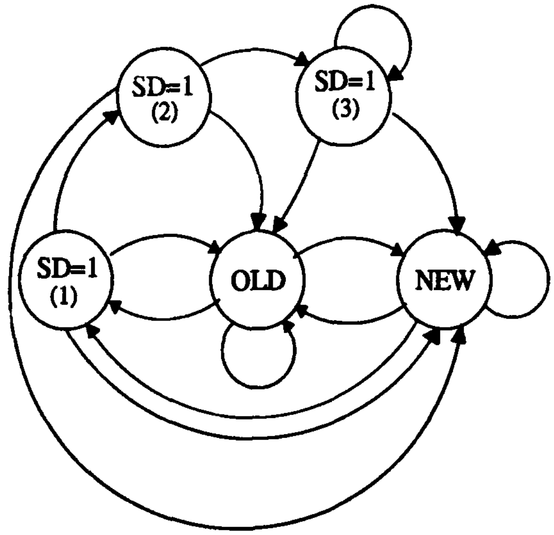

Figure 4.2. Five-State SD=1 Markov Model

a standard deviation of 0.076. This large error is due in large part to the FFT instruction trace where the original string has a third order entropy of 0.30375 and the synthetic string has a third order entropy which averaged 0.60499. In measuring the SD=1 runs for this particular block size it was found that these SD=1 runs have a maximum length of three, with a high percentage of them being of length three. In this five-state model, the average difference in the corresponding third order entropy for block size of 4 in the LISP instruction traces is about the same as the blocks of 8 and 16 which averaged about 0.037. With the LISP traces, these models worked best with the overall traces followed by the instruction traces. For the MIT-ATUM traces, these models did better with the instruction traces followed by the overall traces. As with the original model, the comparison of the third order entropies was poor for the write traces.

4.4.2 Three-State, Four-State, and Five-State SSD Models The second type of model investigated eliminated the SD=1 transition states and added states for the SSD transitions. The previous chapter discussed the effects of blocking the traces on the transition probabilities defined in Hobart's model. With this in mind, the same procedures discussed above were applied to these three models. These models are shown in Figures 4.3, 4.4, and 4.5. They are similar to the SD=1

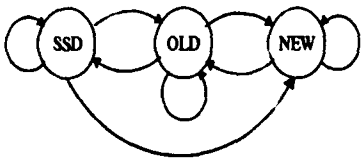

Figure 4.3. Three-State SSD Markov Model

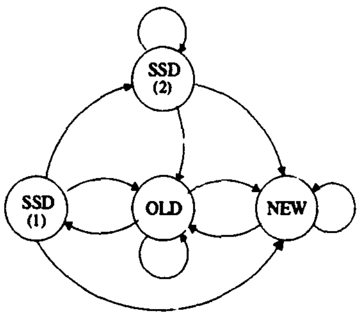

Figure 4.4. Four-State SSD Markov Model

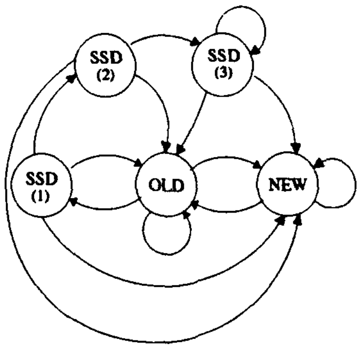

Figure 4.5. Five-State SSD Markov Model

models with SSD states and transitions instead of SD=1 states and transitions. Unlike the SD=1 models, these models did not consider the new references of the original trace to be new references in the blocked trace. In these models, due to the nature of same stack distance, a transition from the new state to the first SSD state cannot occur. The tendency for the first order entropies in these models was to increase with increased block size for all types of traces including the instruction traces. The tendency did not always hold for the overall traces or for second and third order entropies. However, comparing the third order entropy estimation of the original and synthetic strings shows that these models did well with all the types of traces except the instruction traces. The difference in the third order entropy estimation was not always the lowest with the five-state model; but when taken as a percentage of the maximum entropy for the corresponding model with more states, the model provided a closer approximation to the original entropy. Of all the models that were investigated, the five-state SSD model did the best in estimating the third order entropy for the write traces, although it still did not come as close as it did with other types of traces.

4.4.3 Three-State and Four-State SD=1/SSD Model The third and final type of model investigated was a hybrid, exploiting both SD=1 and SSD transitions. These models are shown in Figures 4.6 and 4.7. The four-state model had two separate states for SD=1 and SSD transitions. In this model, the previous old reference was recorded before entering into a run of stack distances equal to one. When a stack distance was encountered which equaled this previous old reference,

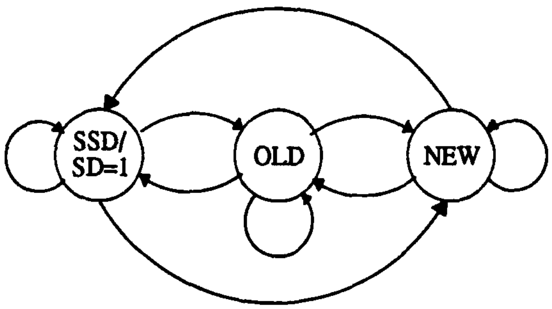

Figure 4.6. Three-State SD=1/SSD Markov Model

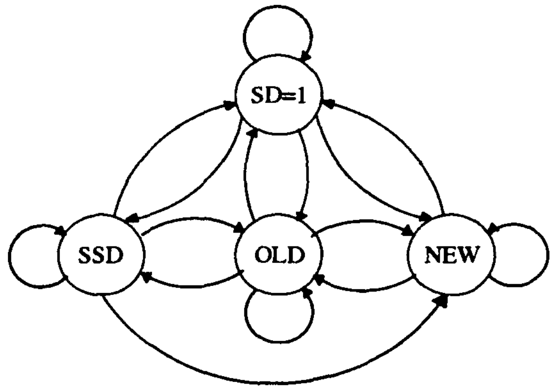

Figure 4.7. Four-State SD=1/SSD Markov Model

a transition was made to the SSD state. Other SD=1 and SSD runs were handled in the same manner as the previous two types of models. In the three-state model the separate SD=1 and SSD states are combined into a single state. For both models, new references are handled in the same way as the SD=1 models by keeping the original new references as transitions to the new state. Of the five types of traces, the three state SD=1/SSD model was closest in estimating the third order entropy of the instruction traces. These models did not do as well with the third order entropy estimation as did the SSD models for the other types of traces. The four-state model did not do as well as three-state model did with every type of trace. The three-state SD=1/SSD model did the best of all the models with the instruction traces.

4.5 Other Considerations

The error in third order entropy estimations is shown in Figures 4.8-4.9 for the eight different models which were previously discussed. These graphs show how the same stack distance models worked well for the all, data, and read reference type traces. It also shows the three-state hybrid model which worked well for the instruction traces from both sets of workloads. The graphs of the write references, particularly from the LISP workloads, demonstrate the difficulty in modeling these types of traces.

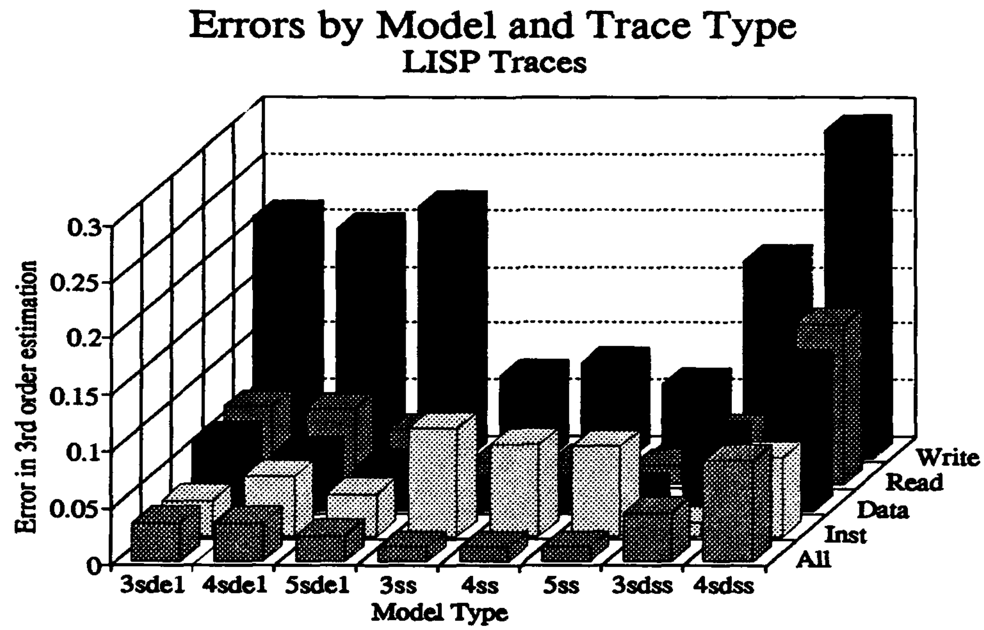

Figure 4.8. Errors in 3rd order entropy estimation for LISP traces.

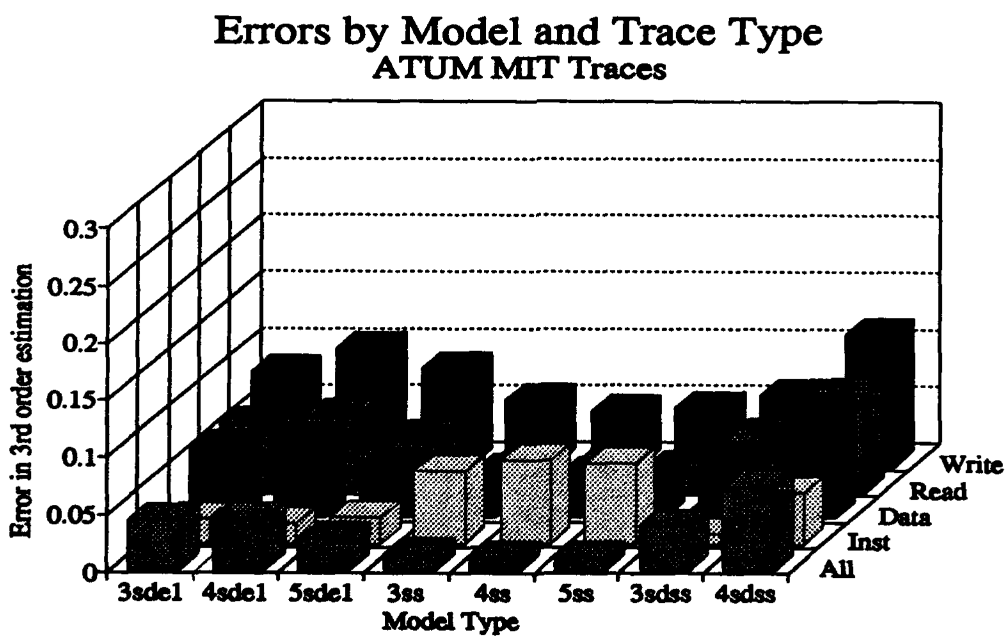

Figure 4.9. Errors in 3rd order entropy estimation for ATUM MIT traces.

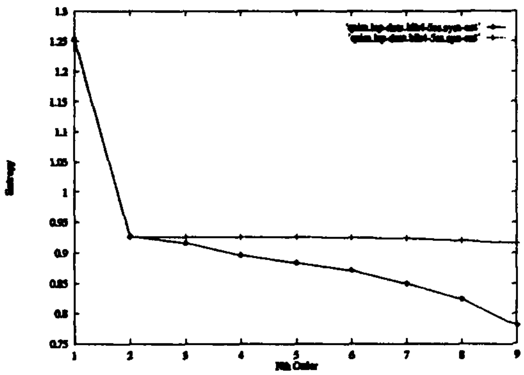

Figure 4.10. Five-State SSD Model Higher Order Entropies

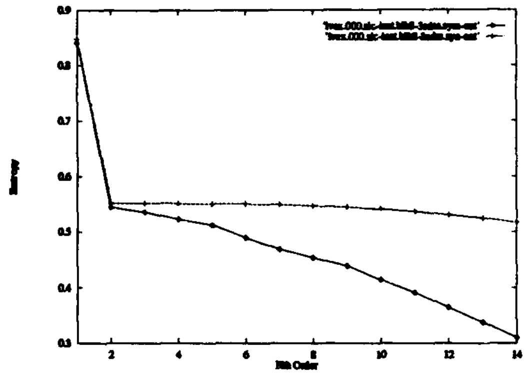

Figure 4.11. Three-State SD=1/SSD Model Higher Order Entropies

In all of the models which were analyzed, the higher order entropies (fourth and greater) of the original string of symbols would continually decrease as the order increased, whereas the higher order entropies for the synthetic string of symbols would quickly become asymptotic with the second order entropy estimation. Figure 4.10 is an example of the five-state SSD model comparing the higher order entropies of the original and a synthetic string for the LISP data trace QSIM using a block size of 4 words. Figure 4.11 is an example of the three-state combined SD=1/SSD model used for the instruction traces. This figure compares the higher order entropies of the original and a synthetic string for the MIT-ATUM instruction trace IVEX.000 using a block size of 8 words. Both plots show how the first and second order entropy estimations for the two strings match up and how the synthetic string fails to contain the predictability that the original string indicates is present. In some cases, higher order entropies of the synthetic string would begin to drop around the 7th or 8th order entropy estimation; however, the entropy of the original string would always decrease much faster.

Due to the recursive nature of entropy estimation (higher order entropy estimations depend on the previous order entropy estimation), once an error is introduced as in the case with the third order entropy estimation, it carries through to the subsequent orders. With each subsequent higher

order entropy estimation, the previous error as well as the error in computing the summation using the particular order N-grams combine to produce an error which grows.

The continued decrease in the entropy of the original string indicates that there exists even more predictability (less randomness) in the sequence of memory address references. The failure of the models to capture these higher order entropies may be due to the models' inability to capture longer run lengths, whether they be SSD or SD=1. Another reason may be the models are not generating a recurring sequence of states which are found in the original string. A sequence such as this may be:

Old, Old, Old, New, Old, Old, Old, New, Old, Old, Old, New ...

From the models' point of view the following sequence would account for the same transition probabilities.

Old, Old, Old, Old, Old, New, Old, Old, New, Old, Old, New ...

The latter case is an example of a pattern which is less predictable than the former case. Several traces were found to have SD=1 runs of length three or less. The 5 state SD=1 model was able to capture all of these runs. However, in looking at the higher order entropy estimations for these traces, even the third order entropy estimation of the corresponding synthetic trace was much higher than that of the original trace. Other patterns are not being generated by the models.

The models' perspective of phases and transitions relies on phases occurring on the old side of the model while transitions occur on the new side. Phases in which the transition probabilities may change are not accounted for since these models assume the transition probabilities are stationary throughout the sequence of references. This assumption can be checked by looking at the entropy within various intervals within the trace. Intervals which have the same behavior would have the

same value for the entropy. A change in entropy indicates a change in the behavior or pattern of references. In some traces, the entropy did in fact remain constant, while in others the entropy would change between intervals. In the cases in which the entropy would remain constant throughout the original trace, the model did not predict the third order entropy estimation any better than in cases in which the entropy did change. However, a model in which the transition probabilities would change with various intervals in the trace would perhaps produce a synthetic string which more closely resembles the original string.

4.6 A Unified Model and Application to Memory Subsystem Design

In building a unified model, the temporal locality of the trace also needs to be considered. Incorporating this can be accomplished by using the P_{LRU} probabilities discussed in the previous section and applying them to the various transition probabilities on the old side of the model. As discussed in Chapter 3, P_{LRU} can be used to determine if a reference will be contained in a fully associative cache using LRU replacement. Temporal locality would be what determines whether an old reference is present somewhere in the memory hierarchy.

4.7 Summary

This chapter has shown how entropy estimation can be used to compare the original reference string with a synthetically generated one. Several versions of different models were introduced exploiting various combinations of SD=1 and SSD transitions. The five state SSD model worked best with all types of traces except for the instruction traces. These traces were modeled best by a combination three-state SD=1/SSD model. All the models were poor in their modeling of the write traces. These type of traces are irregular in their behavior, while the instruction traces are unique in their behavior due to the large percentage of sequential references discussed in chapter 3. Other considerations have been given as to the accuracy of this model in generating the sequence of addresses generated by a computer program.

V. Conclusions

5.1 Contributions of this Research

The principal contribution of this research is the characterization of the relationships between the three types of localities (temporal, spatial, and structural). Temporal locality is able to work with both spatial and structural locality, while spatial and structural locality appear to compete with one another. The exploitation of spatial locality is obtained at the expense of structural locality, while the exploitation of structural locality inhibits exploiting spatial locality.

The existence of temporal locality was shown using the P_{LRU} metric. Its relationship with spatial locality can be seen by comparing the P_{LRU} of blocked versions of the traces which take advantage of spatial locality. The exploitation of spatial locality generally has a positive effect on temporal locality depending on the trace. Temporal locality at higher thresholds may require more memory to exploit spatial locality than if individual words are referenced.

The structural locality was measured using the P_{SSD} metric. While Hobart showed its presence in the LISP traces taken from the TI Explorer, this research has shown that structural locality is also present in the workloads taken from other environments. Structural locality's relationship to spatial locality can also be seen by comparing the P_{SSD} of blocked versions of the traces. The blocking of the traces resulted in a lowering of the P_{SSD} measure. Same stack distance runs in the blocked traces were much shorter in length and generally greater in number. The differences in the state transition probabilities between the word-level and byte-level traces also supports the notion of spatial and structural locality competing. The byte-level traces, which are at a finer resolution, had higher probabilities for P_{SSD}.

The initial model presented utilized the SD=1 transitions which were found to be prevalent in the blocked traces which took advantage of the spatial locality present. Another version of the model capitalized on the same stack distance transitions indicating the presence of structural

locality. Finally, hybrid models which incorporated both SD=1 and SSD transitions were also presented.

The various Markov models used to sythetically generate traces showed that the behavior of the five different types of traces (overall, instruction, data, data read, and data write) did in fact differ. The instruction traces were best synthesized using a model which accounted for both SSD and SD=1 transitions. The overall, data, and data read traces were best synthesized using a model which exploited the same stack distance. The most difficult traces to model were the data write traces. Their unique behavior was also best modeled by using the same stack distance model. The entropy measurement used to compare the original trace with the synthetically generated one indicated that more predictability exists in the original trace than was captured by the model.

5.2 Suggestions for Further Research

The use of longer traces from a variety of environments and workloads is necessary to better see how the different types of localities are affected. Longer traces will ensure that various phases and transitions are covered, while a variety of environments and workloads will provide a better understanding of which characteristics are common and which are peculiar to a given environment, workload, or the program itself. Longer traces will affect the stationarity of the models, and this will need to be accounted for. Studying the source code for these programs may also prove useful.

The differences in behavior between the instruction and data traces warrants the further study of split or Harvard architectures. The P_{SSD} metric used to measure structural locality was high in the instruction references of all workload sets. Within the data references, the reads and writes also have differing behavior. Using source code and knowledge of the data structures being traced will help to find how these types of references are related and why write references are peculiar.

While structural locality has been measured using the P_{SSD} metric, structural locality may actually occur in other ways. Two methods suggested by Bletzinger in [Ble92], reverse access and

transverse access, provide other perspectives of structural locality in data references. Expanding the definition of structural locality to include these types of access behavior can provide a better understanding of its presence and methods to utilize this behavior in increasing the predictability of memory references.

The most important area in which to continue this research is to apply the models and use the characteristics of the traces to predict the performance of a cache/memory subsystem. A cache/memory hierarchy such as Hobart's in [Hob89] can be used and refined to specifically exploit a certain type of locality at given levels in the hierarchy. Due to the competing nature of structural and spatial locality, spatial locality could be exploited at a lower level, while structural locality can be exploited at a higher level. This coincides with the granularity required to exploit these localities. The coarseness of pages in memory management strategies may be one reason that structural locality has not previously been characterized. By comparing the results of trace-driven simulations of various cache architectures and the predictions of an analytical model, the contribution of various types of locality in the program's behavior can be attributed.

Appendix A. *Trace Data*

Table A.1. Dynamic Trace Data

TRACE NAME	NUMBER OF REFS	INST	DATA	READ	WRITE	% Inst	% Read
biaslisp	450000	56448	393552	284272	109280	0.125	0.722
boyer	450000	175400	274600	239059	35541	0.390	0.871
compile-rb	450000	154840	295160	256763	38397	0.344	0.870
compile-str	450000	159769	290231	254356	35875	0.355	0.876
fft	450000	57551	392449	277968	114481	0.128	0.708
glisp-comp	450000	148793	301207	273789	27418	0.331	0.909
glisp-pay	450000	215396	234604	211647	22957	0.479	0.902
qsim	450000	201965	248035	233454	14581	0.449	0.941
reducer	450000	229598	220402	185292	35110	0.510	0.841
tmycin	450000	99529	350471	316096	34375	0.221	0.902
dec0.000	361982	183023	178959	106459	72500	0.506	0.595
fora.000	387934	199799	188135	108979	79156	0.515	0.579
forf.003	368212	190915	177297	107969	69328	0.518	0.609
fsxzz.000	239334	123229	166105	78265	37840	0.515	0.471
ivex.000	341968	203510	138458	97335	41123	0.595	0.703
linp.000	404281	201855	202426	183250	19176	0.499	0.905
lisp.000	291390	169786	121604	99080	22524	0.583	0.815
macr.000	342828	188702	154126	96904	57222	0.550	0.629
memxx.000	444849	219050	225799	126660	99139	0.492	0.561
pasc.000	422090	193025	229065	123708	105357	0.457	0.540
savec.003	228492	139615	88877	73217	15660	0.611	0.824
spic.000	446701	223706	222995	136316	86679	0.501	0.611
ue02.000	357810	199973	157837	98385	59452	0.559	0.623
dec0.001	334775	170283	164492	99897	64595	0.509	0.607
dec1.001	329613	167795	161818	99342	62476	0.509	0.614
dia0	336093	196890	139203	90819	48384	0.586	0.652
forl.000	314110	155713	158397	100019	58378	0.496	0.631
forl.001	362518	169902	192616	116460	76156	0.469	0.605
ivex.000 (dup)	307172	179727	127445	97238	30207	0.585	0.763
ivex.003	396775	171367	225408	137022	88386	0.432	0.608
lisp.000 (dup)	262760	147233	115527	99067	16460	0.560	0.858
lisp.001	261451	146227	115224	98600	16624	0.559	0.856
pasc.001	540567	180020	360547	264969	95578	0.333	0.735
spic.000 (dup)	358168	149832	208336	136088	72248	0.418	0.653
spic.001	442818	172697	250121	151510	98611	0.390	0.606
umil1	357132	171817	185315	167328	17987	0.481	0.903
umil2	359462	163251	196211	182390	13821	0.454	0.930
cc1	1000002	757341	242661	159631	83030	0.757	0.658
spice	1000001	782764	217237	150699	66538	0.783	0.694
tex	832476	597308	235168	130655	104513	0.718	0.556

Table A.2. Static Trace Data

TRACE NAME	percent unique	UNIQUE REFS	INST	DATA	READ	WRITE
biaslisp	0.071	31744	649	31095	30473	28332
boyer	0.045	20283	54	20229	15605	19017
compile-rb	0.039	17475	6708	10771	10390	3666
compile-str	0.045	20027	7070	12962	12377	5474
fft	0.087	38946	110	38836	36784	34126
glisp-comp	0.032	14545	3532	11013	10234	6226
glisp-pay	0.018	8250	758	7493	3955	5381
qsim	0.025	11426	1967	9460	8869	5943
reducer	0.042	18721	1532	17190	16923	15406
tmycin	0.025	11340	1269	10073	8090	6784
dec0.000	0.052	18827	7276	11719	9776	5019
fora.000	0.054	20767	8716	12173	9473	8730
forf.003	0.082	30137	14123	16241	13500	9597
fsxzz.000	0.101	24098	7184	16991	15206	4237
ivex.000	0.108	37033	15210	22091	20227	6789
linp.000	0.031	12456	1974	10513	10273	2220
lisp.000	0.020	5950	929	5038	3788	2964
macr.000	0.070	23972	10343	13800	10488	10997
memxx.000	0.060	26519	7040	19560	5749	16561
pasc.000	0.034	14220	5150	9147	6355	4566
savec.003	0.033	7468	4785	2740	2410	1289
spic.000	0.021	9199	3304	5936	5108	3191
ue02.000	0.088	31612	14559	17315	12589	10753
dec0.001	0.018	6030	2601	3449	3102	859
dec1.001	0.028	9297	4903	4421	3930	1397
dia0	0.037	12425	8636	3812	3542	1034
forl.000	0.052	16189	6849	9375	7366	5156
forl.001	0.044	15980	8874	7126	5639	5297
ivex.000 (dup)	0.103	31517	12443	19127	17893	4703
ivex.003	0.021	8178	4043	4167	3263	2026
lisp.000 (dup)	0.022	5678	784	4895	3693	2813
lisp.001	0.026	6833	835	5999	4366	3088
pasc.001	0.036	19256	1531	17809	13341	17479
spic.000 (dup)	0.022	7710	2845	4886	4172	2851
spic.001	0.013	5584	753	4835	4446	3125
umil1	0.032	11516	6793	4735	4352	882
umil2	0.006	2233	453	1780	1498	331
cc1	0.043	43051	31195	11856	7374	8365
spice	0.015	15320	8964	6356	4227	4624
tex	0.046	38184	159	38026	8214	29827

Table A.3. Spatial Window Probabilities - All References

TRACE NAME	PSW-all	PSW-on	PSW-nn
biaslisp	0.31488	0.02806	0.93963
boyer	0.45189	0.01121	0.95371
compile-rb	0.46762	0.26271	0.69079
compile-str	0.46387	0.26455	0.70903
fft	0.35000	0.00397	0.83968
glisp-comp	0.52833	0.14129	0.74177
glisp-pay	0.44323	0.18893	0.87722
qsim	0.46069	0.13045	0.82607
reducer	0.52534	0.13542	0.96499
tmycin	0.61592	0.07613	0.90095
Mean	0.46218	0.12427	0.84438
Std Dev	0.08614	0.09563	0.10146
dec0.000	0.39797	0.04291	0.53329
fora.000	0.39981	0.05800	0.52034
forf.003	0.35902	0.05385	0.53501
fsxzz.000	0.32709	0.03937	0.33151
ivex.000	0.36103	0.03941	0.55588
linp.000	0.09095	0.00707	0.44523
lisp.000	0.28411	0.01397	0.74768
macr.000	0.36018	0.07792	0.56242
memxx.000	0.43314	0.02458	0.60183
pasc.000	0.42033	0.03918	0.59437
savec.003	0.56863	0.06044	0.58198
spic.000	0.50894	0.05332	0.50706
ue02.000	0.35831	0.05862	0.57166
Mean	0.37458	0.04374	0.54525
Std Dev	0.11336	0.01984	0.09446
dec0.001	0.46675	0.04780	0.54550
dec1.001	0.45970	0.05916	0.55253
dia0	0.34985	0.09761	0.55630
forl.000	0.36836	0.03787	0.59079
forl.001	0.36925	0.07117	0.63585
ivex.000 (dup)	0.33850	0.03524	0.58465
ivex.003	0.40317	0.04382	0.51854
lisp.000 (dup)	0.24392	0.01013	0.76680
lisp.001	0.24254	0.00812	0.77205
pasc.001	0.35307	0.00578	0.92838
spic.000 (dup)	0.41980	0.06275	0.50654
spic.001	0.39319	0.00910	0.26434
umil1	0.14056	0.04831	0.56196
umil2	0.09689	0.01937	0.37069
Mean	0.33182	0.03973	0.58249
Std Dev	0.11169	0.02746	0.16450
cc1	0.60881	0.05857	0.88039
spice	0.62890	0.04509	0.84709
tex	0.51254	0.05882	0.68548
Mean	0.58342	0.05416	0.80432
Std Dev	0.06220	0.00786	0.10426

Table A.4. Spatial Window Probabilities - Inst References

TRACE NAME	PSW-all	PSW-on	PSW-nn
biaslisp	0.91803	0.81250	0.94500
boyer	0.95568	1.00000	0.92683
compile-rb	0.86252	0.72105	0.93156
compile-str	0.87801	0.72087	0.93066
fft	0.99868	1.00000	0.93204
glisp-comp	0.89240	0.72472	0.90424
glisp-pay	0.90083	0.69863	0.93275
qsim	0.90612	0.69231	0.89601
reducer	0.97620	0.67000	0.93014
tmycin	0.88003	0.74126	0.90933
Mean	0.91685	0.77813	0.92386
Std Dev	0.04535	0.12282	0.01534
dec0.000	0.85101	0.39341	0.88818
fora.000	0.86976	0.55034	0.88324
forf.003	0.84439	0.47864	0.87645
fsxzz.000	0.86424	0.51136	0.89328
ivex.000	0.88040	0.40662	0.88211
linp.000	0.96115	0.50000	0.87053
lisp.000	0.84549	0.39655	0.88621
macr.000	0.87127	0.41534	0.88433
memxx.000	0.93172	0.58841	0.88439
pasc.000	0.84115	0.33103	0.85700
savec.003	0.96734	0.54167	0.89699
spic.000	0.88529	0.43478	0.88289
ue02.000	0.86812	0.40396	0.89469
Mean	0.88318	0.45785	0.88310
Std Dev	0.04298	0.07614	0.01062
dec0.001	0.93797	0.42593	0.92051
dec1.001	0.93505	0.50211	0.90698
dia0	0.85241	0.61294	0.90635
forl.000	0.87823	0.56044	0.90925
forl.001	0.90554	0.62311	0.92977
ivex.000 (dup)	0.91873	0.56382	0.91465
ivex.003	0.89504	0.62257	0.91627
lisp.000 (dup)	0.86235	0.50725	0.92577
lisp.001	0.86228	0.51389	0.92529
pasc.001	0.96807	0.63953	0.90235
spic.000 (dup)	0.87177	0.50628	0.90599
spic.001	0.90120	0.46000	0.90171
umil1	0.87006	0.58557	0.90721
umil2	0.86475	0.56604	0.91500
Mean	0.89453	0.54925	0.91336
Std Dev	0.03487	0.06518	0.00914
cc1	0.91200	0.46898	0.93736
spice	0.88437	0.34491	0.94983
tex	0.90624	0.50000	0.90260
Mean	0.90087	0.43796	0.92993
Std Dev	0.01458	0.08207	0.02448

Table A.5. Spatial Window Probabilities - Data References

TRACE NAME	PSW-all	PSW-on	PSW-nn
biaslisp	0.32138	0.02718	0.96228
boyer	0.21535	0.01044	0.95403
compile-rb	0.37716	0.29106	0.69716
compile-str	0.39357	0.29233	0 72753
fft	0.33588	0.00386	0.80788
glisp-comp	0.43030	0.12551	0.71863
glisp-pay	0.33421	0.17690	0.89411
qsim	0.56246	0.14748	0.81903
reducer	0.56211	0.08622	0.96612
tmycin	0.63720	0.06207	0.90211
Mean	0.41696	0.12231	0.84489
Std Dev	0.13176	0.10621	0.10523
dec0.000	0.52768	0.08366	0.62279
fora.000	0.50067	0.10642	0.59671
forf.003	0.47789	0.12568	0.55289
fsxzz.000	0.58860	0.05952	0.22400
ivex.000	0.38560	0.05787	0.62184
linp.000	0.11183	0.00580	0.42339
lisp.000	0.29911	0.01883	0.84404
macr.000	0.41031	0.10498	0.55809
memxx.000	0.49226	0.04165	0.69663
pasc.000	0.59181	0.07496	0.70185
savec.003	0.79789	0.12043	0.63430
spic.000	0.60097	0.09715	0.57494
ue02.000	0.45797	0.10834	0.64143
Mean	0.48020	0.07733	0.59176
Std Dev	0.14362	0.09267	0.22821
dec0.001	0.59021	0.26732	0.42444
dec1.001	0.58487	0.29512	0.52855
dia0	0.55019	0.24864	0.55874
forl.000	0.48044	0.07395	0.67805
forl.001	0.55806	0.13218	0.65742
ivex.000 (dup)	0.37959	0.05768	0.68748
ivex.003	0.54196	0.07692	0.56750
lisp.000 (dup)	0.32446	0.01956	0.86783
lisp.001	0.33312	0 02084	0.87285
pasc.001	0.59444	0.00601	0.97932
spic.000 (dup)	0.61720	0.12328	0.61268
spic.001	0.57593	0.07906	0.41623
umil1	0.10574	0.22623	0.55987
umil2	0.04672	0.01077	0.39932
Mean	0.44878	0.11697	0.62931
Std Dev	0.18587	0.10176	0.17750
cc1	0.44107	0.09062	0.60150
spice	0.48233	0.05815	0.80523
tex	0.61028	0.00004	0.00084
Mean	0.51123	0.04960	0.46919
Std Dev	0.08823	0.04589	0.41820

Table A.6. Spatial Window Probabilities - Read References

TRACE NAME	PSW-all	PSW-on	PSW-nn
biaslisp	0.28736	0.38873	0.85687
boyer	0.17804	0.46689	0.69643
compile-rb	0.33123	0.31369	0.64204
compile-str	0.35269	0.33517	0.68455
fft	0.30362	0.38117	0.68049
glisp-comp	0.40661	0.26523	0.67069
glisp-pay	0.28710	0.17509	0.64311
qsim	0.56376	0.38701	0.74496
reducer	0.49342	0.05444	0.72078
tmycin	0.64541	0.16369	0.78858
Mean	0.38492	0.29311	0.71285
Std Dev	0.14318	0.12760	0.06778
dec0.000	0.42520	0.10228	0.59621
fora.000	0.38109	0.13223	0.57807
forf.003	0.38250	0.14040	0.53882
fsxzz.000	0.47601	0.06479	0.20344
ivex.000	0.29208	0.05516	0.62414
linp.000	0.05250	0.00487	0.34850
lisp.000	0.24304	0.01054	0.51159
macr.000	0.34663	0.07692	0.68950
memxx.000	0.27220	0.20592	0.66027
pasc.000	0.48996	0.17290	0.76820
savec.003	0.79462	0.14503	0.62796
spic.000	0.50041	0.10183	0.77640
ue02.000	0.39260	0.11742	0.64899
Mean	0.38837	0.10233	0.58247
Std Dev	0.17212	0.05938	0.15900
dec0.001	0.41321	0.30211	0.38142
dec1.001	0.41367	0.32612	0.48459
dia0	0.40318	0.17468	0.49335
forl.000	0.43169	0.12402	0.78878
forl.001	0.45842	0.18541	0.52230
ivex.000 (dup)	0.32163	0.06515	0.67574
ivex.003	0.43808	0.15866	0.73929
lisp.000 (dup)	0.29067	0.01048	0.55882
lisp.001	0.29946	0.00913	0.62559
pasc.001	0.58350	0.00806	0.97134
spic.000 (dup)	0.54338	0.16094	0.80502
spic.001	0.56218	0.22760	0.91218
umil1	0.06993	0.23267	0.52994
umil2	0.02531	0.00408	0.36131
Mean	0.37531	0.14208	0.63212
Std Dev	0.16543	0.10983	0.18942
cc1	0.38411	0.11814	0.63905
spice	0.46186	0.13930	0.90047
tex	0.72733	0.00098	0.99951
Mean	0.52443	0.08614	0.84634
Std Dev	0.17996	0.07451	0.18623

Table A.7. Spatial Window Probabilities - Write References

TRACE NAME	PSW-all	PSW-on	PSW-nn
biaslisp	0.23081	0.00338	0.91843
boyer	0.38631	0.00042	0.99832
compile-rb	0.44610	0.05765	0.95761
compile-str	0.46832	0.04820	0.95556
fft	0.17455	0.00046	0.79928
glisp-comp	0.37844	0.02351	0.97021
glisp-pay	0.62264	0.00701	0.98179
qsim	0.42706	0.00140	0.97604
reducer	0.89613	0.00404	0.98871
tmycin	0.46089	0.01062	0.98717
Mean	0.44913	0.01567	0.95331
Std Dev	0.20081	0.02091	0.05869
dec0.000	0.68459	0.18041	0.82007
fora.000	0.67884	0.13833	0.71229
forf.003	0.65813	0.12614	0.53454
fsxzz.000	0.75460	0.16187	0.77468
ivex.000	0.55553	0.15691	0.71508
linp.000	0.65665	0.03971	0.21949
lisp.000	0.46231	0.03955	0.89142
macr.000	0.46461	0.07200	0.55360
memxx.000	0.49743	0.02719	0.81895
pasc.000	0.80185	0.17249	0.87422
savec.003	0.76756	0.24895	0.69630
spic.000	0.73656	0.21157	0.84154
ue02.000	0.56005	0.15561	0.74869
Mean	0.63682	0.13313	0.70776
Std Dev	0.11729	0.06938	0.18322
dec0.001	0.92123	0.38272	0.90894
dec1.001	0.91096	0.38136	0.91363
dia0	0.83075	0.52344	0.88295
forl.000	0.73536	0.12075	0.69637
forl.001	0.79169	0.17476	0.84262
ivex.000 (dup)	0.72235	0.23036	0.83322
ivex.003	0.80634	0.14619	0.89595
lisp.000 (dup)	0.54107	0.05192	0.94097
lisp.001	0.53218	0.05801	0.92803
pasc.001	0.62321	0.00640	0.75171
spic.000 (dup)	0.83161	0.22639	0.84733
spic.001	0.77329	0.30964	0.72791
umil1	0.48029	0.32290	0.87297
umil2	0.24456	0.06286	0.82581
Mean	0.69606	0.21412	0.84774
Std Dev	0.19004	0.15318	0.07577
cc1	0.59950	0.24728	0.64527
spice	0.59646	0.04222	0.87500
tex	0.84615	0.00000	0.99991
Mean	0.68070	0.09650	0.84006
Std Dev	0.14329	0.13227	0.17988

Table A.8. Temporal Locality Probabilities - All References

TRACE NAME all	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold
biaslisp	4.697	1491	90.000	79.647	25283	95.004	82.142	26075	99.001
boyer	1.095	222	90.050	4.132	838	95.004	19.647	3985	99.001
compile-rb	2.678	468	90.030	4.853	848	95.004	19.187	3353	99.001
compile-str	5.123	1026	90.013	11.010	2205	95.011	83.862	16795	99.000
fft	13.948	5432	90.001	74.816	29138	95.041	74.976	29200	99.019
glisp-comp	5.782	841	90.019	9.687	1409	95.025	31.255	4546	99.000
glisp-pay	19.855	1638	90.038	20.279	1673	95.235	20.545	1695	99.107
qsim	4.691	536	90.001	12.787	1461	95.014	37.546	4290	99.001
reducer	2.778	520	90.102	4.321	809	95.000	14.048	2630	99.001
tmycin	5.608	636	90.026	9.162	1039	95.031	21.975	2492	99.000
dec0.001	10.100	609	90.017	19.038	1148	95.002	49.917	3010	99.011
dec1.001	7.669	713	90.020	15.639	1454	95.006	45.144	4197	99.000
dia0	6.318	785	90.019	17.352	2156	95.000	70.350	8741	99.000
forl.000	9.636	1560	90.020	12.323	1995	95.001	26.141	4232	99.037
forl.001	4.944	790	90.004	9.287	1484	95.003	27.991	4473	99.000
ivex.000	3.985	1256	90.003	11.178	3523	95.000	54.602	17209	99.002
ivex.003	17.144	1402	90.079	19.467	1592	95.005	28.026	2292	99.083
lisp.000	6.992	397	90.007	11.483	652	95.002	31.789	1805	99.000
lisp.001	5.810	397	90.056	8.635	590	95.006	27.792	1899	99.002
pasc.001	0.576	111	90.086	7.359	1417	95.388	39.878	7679	99.001
spic.000	31.440	2424	90.027	39.261	3027	95.000	56.459	4353	99.004
spic.001	7.468	417	90.006	42.872	2394	95.001	75.985	4243	99.001
umil1	3.352	386	90.002	7.259	836	95.001	42.810	4930	99.006
umil2	13.927	311	91.309	21.272	475	95.004	44.559	995	99.146

Table A.9. Temporal Locality Probabilities - Inst References

TRACE NAME inst	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold
biaslisp	55.932	363	97.054	55.932	363	97.054	88.906	577	99.118
boyer	55.556	30	93.213	61.111	33	95.300	92.593	50	99.362
compile-rb	4.025	270	90.016	6.768	454	95.048	20.081	1347	99.090
compile-str	7.963	563	90.200	17.298	1223	95.031	95.389	6744	99.001
fft	26.364	29	96.993	26.364	29	96.993	36.364	40	99.119
glisp-comp	13.165	465	90.087	18.573	656	95.076	34.655	1224	99.009
glisp-pay	99.340	753	97.693	99.340	753	97.693	99.736	756	99.966
qsim	6.914	136	90.409	30.097	592	95.373	32.283	635	99.020
reducer	4.700	72	90.708	10.248	157	95.113	44.125	676	99.002
tmycin	22.931	291	90.186	40.110	509	95.070	50.276	638	99.013
dec0.001	15.417	401	90.008	22.684	590	95.007	49.366	1284	99.011
dec1.001	8.770	430	90.002	13.359	655	95.035	47.746	2341	99.005
dia0	7.295	630	90.008	26.725	2308	95.036	75.035	6480	99.000
forl.000	16.703	1144	90.024	21.667	1484	95.002	35.699	2445	99.117
forl.001	6.232	553	90.004	8.902	790	95.003	15.416	1368	99.051
ivex.000	4.669	581	90.037	13.461	1675	95.003	59.535	7408	99.003
ivex.003	26.441	1069	90.796	28.840	1166	95.012	40.589	1641	99.019
lisp.000	24.362	191	90.064	36.352	285	95.008	61.097	479	99.235
lisp.001	22.635	189	90.870	32.096	268	95.016	52.934	442	99.033
pasc.001	4.050	62	94.661	6.662	102	95.000	14.827	227	99.128
spic.000	63.480	1806	90.058	71.529	2035	95.208	74.868	2130	99.094
spic.001	7.171	54	90.018	9.429	71	96.756	55.378	417	99.000
umil1	2.444	166	90.307	9.068	616	95.026	46.195	3138	99.003
umil2	21.854	99	91.163	27.152	123	95.013	48.786	221	99.009

Table A.10. Temporal Locality Probabilities - Data References

TRACE NAME data	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold
biaslisp	44.856	13948	90.000	79.730	24792	95.006	82.287	25587	99.004
boyer	3.446	697	90.003	5.299	1072	95.000	25.404	5139	99.001
compile-rb	2.135	230	90.113	3.807	410	95.000	20.750	2235	99.001
compile-str	3.410	442	90.023	8.548	1108	95.002	76.663	9937	99.001
fft	42.813	16627	90.000	74.779	29041	95.065	74.915	29094	99.050
glisp-comp	4.213	464	90.075	7.391	814	95.002	32.262	3553	99.001
glisp-pay	12.145	910	90.036	12.305	922	95.252	12.585	943	99.064
qsim	6.025	570	90.001	9.228	873	95.005	37.357	3534	99.001
reducer	2.757	474	90.007	4.316	742	95.003	11.193	1924	99.001
tmycin	4.249	428	90.054	5.748	579	95.005	17.194	1732	99.004
dec0.001	7.567	261	90.015	18.962	654	95.003	63.729	2198	99.000
dec1.001	7.487	331	90.003	17.869	790	95.015	45.895	2029	99.037
dia0	5.299	202	90.085	12.408	473	95.000	53.620	2044	99.001
forl.000	3.307	310	90.010	5.792	543	95.026	16.992	1593	99.000
forl.001	3.929	280	90.003	10.806	770	95.002	47.334	3373	99.002
ivex.000	3.953	756	90.005	14.242	2724	95.002	55.100	10539	99.001
ivex.003	4.560	190	90.062	9.671	403	95.103	18.599	775	99.002
lisp.000	4.699	230	90.030	12.748	624	95.013	34.484	1688	99.028
lisp.001	3.651	219	90.017	7.901	474	95.003	27.588	1655	99.022
pasc.001	1.174	209	90.005	7.232	1288	95.813	52.473	9345	99.097
spic.000	5.158	252	90.002	19.505	953	95.000	46.009	2248	99.001
spic.001	31.913	1543	90.001	55.367	2677	95.000	79.504	3844	99.002
umil1	4.710	223	90.006	9.884	468	95.074	38.268	1812	99.002
umil2	12.022	214	90.891	26.067	464	95.138	43.708	778	99.120

Table A.11. Temporal Locality Probabilities - Read References

TRACE NAME read	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold
biaslisp	77.839	23720	90.013	79.316	24170	95.006	82.119	25024	99.003
boyer	3.460	540	90.038	4.774	745	95.003	17.353	2708	99.002
compile-rb	2.281	237	90.036	4.158	432	95.015	22.474	2335	99.001
compile-str	3.862	478	90.005	9.405	1164	95.000	78.298	9691	99.002
fft	66.888	24604	90.000	73.407	27002	95.003	73.505	27038	99.055
glisp-comp	4.544	465	90.025	8.403	860	95.034	33.555	3434	99.027
glisp-pay	21.568	853	90.253	21.795	862	95.086	22.276	881	99.044
qsim	6.900	612	90.009	9.798	869	95.021	39.429	3497	99.000
reducer	2.960	501	90.005	4.910	831	95.003	13.821	2339	99.016
tmycin	5.290	428	90.054	7.478	605	95.020	19.518	1579	99.001
dec0.001	14.861	461	90.013	27.047	839	95.016	72.405	2246	99.001
dec1.001	16.463	647	90.020	23.333	917	95.045	59.873	2353	99.011
dia0	7.933	281	90.008	16.177	573	95.003	60.248	2134	99.038
forl.000	5.023	370	90.017	7.453	549	95.011	21.233	1564	99.005
forl.001	6.029	340	90.024	16.475	929	95.003	49.104	2769	99.000
ivex.000	6.589	1179	90.004	17.515	3134	95.000	56.123	10042	99.002
ivex.003	8.091	264	90.006	12.013	392	95.149	24.241	791	99.000
lisp.000	6.309	233	90.012	16.437	607	95.003	37.206	1374	99.003
lisp.001	5.016	219	90.027	11.131	486	95.002	35.891	1567	99.001
pasc.001	0.450	60	90.010	6.424	857	95.328	47.095	6283	99.004
spic.000	14.382	600	90.000	22.747	949	95.004	47.819	1995	99.000
spic.001	41.633	1851	90.001	62.528	2780	95.002	79.667	3542	99.004
umil1	4.021	175	90.205	5.400	235	95.004	36.443	1586	99.001
umil2	11.682	175	92.171	12.884	193	95.237	34.379	515	99.016

Table A.12. Temporal Locality Probabilities - Write References

TRACE NAME write	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold	%of total words	total size (words)	threshold
biaslisp	76.602	21703	90.016	77.488	21954	95.002	77.997	22098	99.099
boyer	0.042	8	90.947	0.068	13	95.909	0.163	31	99.020
compile-rb	1.227	45	90.101	3.873	142	95.099	40.289	1477	99.004
compile-str	2.594	142	90.247	8.531	467	95.010	51.827	2837	99.000
fft	72.997	24911	90.211	73.021	24919	95.453	73.073	24937	99.457
glisp-comp	2.281	142	90.020	5.911	368	95.003	46.916	2921	99.042
glisp-pay	2.044	110	90.487	2.657	143	99.744	2.657	143	99.744
qsim	1.380	82	90.021	1.599	95	95.728	22.076	1312	99.004
reducer	0.526	81	91.301	0.552	85	95.179	9.464	1458	99.031
tmycin	1.474	100	90.120	2.535	172	95.020	12.294	834	99.003
dec0.001	14.319	123	90.018	17.928	154	95.058	28.754	247	99.013
dec1.001	9.019	126	90.064	11.238	157	95.128	21.546	301	99.001
dia0	4.642	48	90.131	7.157	74	95.088	34.139	353	99.010
forl.000	4.480	231	90.002	6.497	335	95.019	16.040	827	99.002
forl.001	3.549	188	90.017	7.457	395	95.004	38.663	2048	99.001
ivex.000	6.974	328	90.017	23.219	1092	95.024	57.602	2709	99.000
ivex.003	11.747	238	90.078	14.511	294	95.060	25.222	511	99.008
lisp.000	8.034	226	90.049	13.224	372	95.039	37.291	1049	99.011
lisp.001	6.574	203	90.056	11.788	364	95.006	36.334	1122	99.003
pasc.001	29.390	5137	90.005	49.682	8684	95.014	58.447	10216	99.356
spic.000	7.646	218	90.010	18.029	514	95.082	39.320	1121	99.001
spic.001	30.848	964	90.004	57.472	1796	95.010	72.768	2274	99.016
umil1	31.859	281	90.552	33.900	299	95.656	94.444	833	99.123
umil2	84.894	281	93.877	85.196	282	95.708	86.707	287	99.807

Table A.13. State Transition Probabilities - All References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.70621	0.26919	0.67722	0.05360	0.29379
boyer	0.50582	0.47390	0.50223	0.02387	0.49418
compile-rb	0.38169	0.56177	0.42282	0.01542	0.61831
compile-str	0.38358	0.54389	0.43825	0.01786	0.61642
fft	0.68528	0.31622	0.61885	0.06493	0.31472
glisp-comp	0.47790	0.62282	0.36122	0.01596	0.52210
glisp-pay	0.24970	0.58841	0.40693	0.00466	0.75030
qsim	0.45751	0.44436	0.54372	0.01192	0.54249
reducer	0.13653	0.54049	0.45358	0.00592	0.86347
tmycin	0.37769	0.62604	0.36420	0.00976	0.62231
Mean	0.43619	0.49871	0.47890	0.02239	0.56381
Std Dev	0.17503	0.12337	0.10608	0.02042	0.17503
dec0.000	0.68327	0.33436	0.62816	0.03748	0.31673
fora.000	0.64842	0.34944	0.61389	0.03667	0.35158
forf.003	0.60565	0.33687	0.60914	0.05399	0.39435
fsxzz.000	0.53760	0.35351	0.58630	0.06019	0.46240
ivex.000	0.66461	0.47406	0.44523	0.08071	0.33539
linp.000	0.85195	0.14023	0.83269	0.02708	0.14805
lisp.000	0.63765	0.38488	0.60183	0.01329	0.36235
macr.000	0.66918	0.38108	0.56861	0.05031	0.33082
memxx.000	0.76563	0.36194	0.58952	0.04853	0.23437
pasc.000	0.63000	0.32765	0.65039	0.02196	0.37000
savec.003	0.52303	0.76882	0.21351	0.01767	0.47697
spic.000	0.56882	0.44050	0.54754	0.01196	0.43118
ue02.000	0.59033	0.35417	0.58862	0.05721	0.40967
Mean	0.64432	0.38519	0.57503	0.03977	0.35568
Std Dev	0.08984	0.13849	0.13758	0.02090	0.08984
dec0.001	0.65915	0.33098	0.65693	0.01209	0.34085
dec1.001	0.56175	0.32995	0.65375	0.01630	0.43825
dia0	0.46181	0.42119	0.56108	0.01773	0.53819
forl.000	0.65400	0.35639	0.60807	0.03554	0.34600
forl.001	0.65680	0.25956	0.71016	0.03029	0.34320
ivex.000 (dup)	0.68962	0.46282	0.45833	0.07884	0.31038
ivex.003	0.55803	0.29398	0.69427	0.01174	0.44197
lisp.000 (dup)	0.64354	0.35492	0.63088	0.01421	0.35646
lisp.001	0.68472	0.35169	0.62994	0.01837	0.31528
pasc.001	0.48512	0.39320	0.58888	0.01792	0.51488
spic.000 (dup)	0.55390	0.34804	0.63978	0.01218	0.44610
spic.001	0.72833	0.29730	0.69295	0.00975	0.27167
umil1	0.56626	0.29924	0.68189	0.01886	0.43374
umil2	0.74026	0.26681	0.72856	0.00462	0.25974
Mean	0.61738	0.34043	0.63825	0.02132	0.38262
Std Dev	0.08658	0.05738	0.07003	0.01835	0.08658
cc1	0.49520	0.52237	0.45535	0.02228	0.50480
spice	0.53557	0.59913	0.39254	0.00833	0.46443
tex	0.99610	0.41821	0.53391	0.04768	0.00390
Mean	0.67562	0.51324	0.46060	0.02616	0.32438
Std Dev	0.27827	0.09081	0.07083	0.02006	0.27827

Table A.14. State Transition Probabilities - Inst References

TRACE NAME	INST P New-Old	INST P SSD	INST P NSSD	INST P Old-New	INST P New-New
biaslisp	0.07550	0.88105	0.11809	0.00086	0.92450
boyer	0.24074	0.56303	0.43690	0.00007	0.75926
compile-rb	0.11345	0.86252	0.13235	0.00513	0.88655
compile-str	0.11669	0.83820	0.15641	0.00540	0.88331
fft	0.06364	0.99502	0.00487	0.00010	0.93636
glisp-comp	0.15147	0.78162	0.21470	0.00368	0.84853
glisp-pay	0.09763	0.83051	0.16915	0.00034	0.90237
qsim	0.15913	0.76726	0.23118	0.00156	0.84087
reducer	0.19648	0.95149	0.04720	0.00132	0.80352
tmycin	0.11348	0.69332	0.30522	0.00146	0.88652
Mean	0.13282	0.81640	0.18161	0.00199	0.86718
Std Dev	0.05483	0.12477	0.12497	0.00201	0.05483
dec0.000	0.07944	0.90316	0.09356	0.00328	0.92056
fora.000	0.08308	0.92717	0.06904	0.00379	0.91692
forf.003	0.06472	0.91798	0.07685	0.00516	0.93528
fsxzz.000	0.04914	0.97272	0.02425	0.00303	0.95086
ivex.000	0.07141	0.96063	0.03360	0.00577	0.92859
linp.000	0.03700	0.98359	0.01604	0.00037	0.96300
lisp.000	0.06351	0.88171	0.11795	0.00034	0.93649
macr.000	0.06043	0.97404	0.02245	0.00350	0.93957
memxx.000	0.04646	0.96831	0.03015	0.00154	0.95354
pasc.000	0.05613	0.91054	0.08792	0.00154	0.94387
savec.003	0.04033	0.98262	0.01595	0.00142	0.95967
spic.000	0.06933	0.93889	0.06007	0.00104	0.93067
ue02.000	0.07968	0.92808	0.06566	0.00626	0.92032
Mean	0.06159	0.94226	0.05488	0.00285	0.93841
Std Dev	0.01514	0.03357	0.03350	0.00200	0.01514
dec0.001	0.12423	0.81788	0.18019	0.00193	0.87577
dec1.001	0.09649	0.82845	0.16865	0.00290	0.90351
dia0	0.09114	0.86868	0.12714	0.00418	0.90886
forl.000	0.07959	0.88813	0.10821	0.00366	0.92041
forl.001	0.11911	0.86458	0.12886	0.00656	0.88089
ivex.000 (dup)	0.09500	0.91877	0.07416	0.00707	0.90500
ivex.003	0.06333	0.90996	0.08851	0.00153	0.93666
lisp.000 (dup)	0.08929	0.85504	0.14449	0.00047	0.91071
lisp.001	0.08513	0.85627	0.14324	0.00049	0.91487
pasc.001	0.05683	0.98721	0.01230	0.00048	0.94317
spic.000 (dup)	0.08368	0.90566	0.09272	0.00162	0.91632
spic.001	0.06773	0.99055	0.00915	0.00029	0.93227
umil1	0.08760	0.88153	0.11486	0.00361	0.91240
umil2	0.11504	0.86727	0.13241	0.00032	0.88496
Mean	0.08959	0.88857	0.10892	0.00251	0.91041
Std Dev	0.02000	0.05117	0.05083	0.00227	0.01999
cc1	0.05273	0.95125	0.04649	0.00226	0.94727
spice	0.04830	0.96507	0.03437	0.00056	0.95170
tex	0.03145	0.94997	0.05002	0.00001	0.96855
Mean	0.04416	0.95543	0.04363	0.00094	0.95584
Std Dev	0.01123	0.00837	0.00821	0.00117	0.01123

Table A.15. State Transition Probabilities - Data References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.71352	0.25914	0.67965	0.06121	0.28648
boyer	0.50648	0.36965	0.59008	0.04028	0.49352
compile-rb	0.45400	0.46432	0.51849	0.01719	0.54600
compile-str	0.44476	0.45479	0.52442	0.02079	0.55524
fft	0.67378	0.28959	0.63641	0.07400	0.32622
glisp-comp	0.53401	0.61561	0.36413	0.02026	0.46599
glisp-pay	0.25891	0.44239	0.54907	0.00854	0.74109
qsim	0.47658	0.64611	0.33499	0.01890	0.52342
reducer	0.12083	0.40696	0.58283	0.01022	0.87917
tmycin	0.39353	0.66893	0.31943	0.01164	0.60647
Mean	0.45764	0.46175	0.50995	0.02830	0.54236
Std Dev	0.17575	0.14267	0.12759	0.02271	0.17575
dec0.000	0.65185	0.16654	0.78779	0.04567	0.34815
fora.000	0.67551	0.15703	0.79624	0.04673	0.32449
forf.003	0.61287	0.19046	0.74775	0.06180	0.38713
fsxzz.000	0.54388	0.28600	0.62078	0 09323	0.45612
ivex.000	0.76035	0.13879	0.71687	0.14434	0.23965
linp.000	0.91867	0.06174	0.88794	0.05032	0.08133
lisp.000	0.65383	0.16608	0.80567	0.02825	0.34617
macr.000	0.77732	0.13625	0.78732	0.07644	0.22268
memxx.000	0.84074	0.23900	0.68126	0.07973	0.15926
pasc.000	0.71619	0.20035	0.76987	0.02978	0.28381
savec.003	0.54891	0.64876	0.33379	0.01745	0.45109
spic.000	0.65044	0.23459	0.74763	0.01778	0.34956
ue02.000	0.57736	0.14655	0.78232	0.07114	0.42264
Mean	0.68676	0.21324	0.72809	0.05867	0.31324
Std Dev	0.18133	0.16910	0.17537	0.02034	0.18840
dec0.001	0.64888	0.15369	0.83242	0.01389	0.35112
dec1.001	0.61185	0.13963	0.84319	0.01718	0.38815
dia0	0.62487	0.16597	0.81644	0.01759	0.37513
forl.000	0.78763	0.15617	0.79429	0.04954	0.21237
forl.001	0.69642	0.14376	0.82949	0.02675	0.30358
ivex.000 (dup)	0.78319	0.14024	0.72147	0.13829	0.21681
ivex.003	0.72090	0.16347	0.82296	0.01357	0.27910
lisp.000 (dup)	0.64760	0.14647	0.82489	0.02864	0.35240
lisp.001	0.68011	0.13941	0.82324	0.03735	0.31989
pasc.001	0.49500	0.42062	0.55366	0.02572	0.50500
spic.000 (dup)	0.60950	0.22229	0.76308	0.01463	0.39050
spic.001	0.72223	0.18660	0.79916	0.01423	0.27777
umil1	0.74424	0.14189	0.83860	0.01951	0.25576
umil2	0.81697	0.15012	0.84312	0.00676	0.18303
Mean	0.68496	0.17645	0.79329	0.03026	0.31504
Std Dev	0.08656	0.07386	0.07682	0.03302	0.08656
cc1	0.53753	0.16576	0.80663	0.02761	0.46247
spice	0.63295	0.23537	0.74556	0.01907	0.36705
tex	0.62257	0.26537	0.61455	0.12008	0.37743
Mean	0.59768	0.22217	0.72225	0.05559	0.40232
Std Dev	0.05235	0.05110	0.09814	0.05602	0.05235

Table A.16. State Transition Probabilities - Read References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.68703	0.46702	0.45049	0.08249	0.31297
boyer	0.89234	0.43184	0.50585	0.06231	0.10766
compile-rb	0.48027	0.60871	0.37104	0.02025	0.51973
compile-str	0.45471	0.57300	0.40374	0.02325	0.54529
fft	0.65319	0.52214	0.37824	0.09962	0.34681
glisp-comp	0.55492	0.70062	0.27784	0.02154	0.44509
glisp-pay	0.57067	0.48421	0.50493	0.01086	0.42933
qsim	0.56940	0.68842	0.28910	0.02248	0.43060
reducer	0.89334	0.49239	0.41783	0.08979	0.10666
tmycin	0.62991	0.74993	0.23353	0.01654	0.37009
Mean	0.63858	0.57183	0.38326	0.04491	0.36142
Std Dev	0.15173	0.11071	0.09342	0.03466	0.15173
dec0.000	0.63318	0.31425	0.62174	0.06401	0.36682
fora.000	0.68426	0.26965	0.66521	0.06513	0.31574
forf.003	0.58878	0.28078	0.63509	0.08413	0.41122
fsxzz.000	0.50349	0.29516	0.58345	0.12140	0.49651
ivex.000	0.76284	0.24968	0.55022	0.20010	0.23716
linp.000	0.91872	0.09333	0.85212	0.05456	0.08128
lisp.000	0.85190	0.25769	0.70845	0.03385	0.14810
macr.000	0.68678	0.26420	0.65246	0.08334	0.31322
memxx.000	0.52896	0.60280	0.37205	0.02514	0.47104
pasc.000	0.51802	0.38787	0.58409	0.02804	0.48198
savec.003	0.49253	0.77458	0.20867	0.01675	0.50747
spic.000	0.58849	0.46328	0.51381	0.02290	0.41151
ue02.000	0.60894	0.25677	0.65388	0.08934	0.39106
Mean	0.64361	0.34693	0.58471	0.06836	0.35639
Std Dev	0.13402	0.17663	0.15796	0.05059	0.13402
dec0.001	0.64249	0.30581	0.67361	0.02058	0.35751
dec1.001	0.61196	0.28671	0.68810	0.02520	0.38804
dia0	0.57554	0.29524	0.68141	0.02335	0.42446
forl.000	0.66128	0.28847	0.65897	0.05256	0.33872
forl.001	0.70965	0.26926	0.69463	0.03610	0.29035
ivex.000 (dup)	0.79869	0.24414	0.57576	0.18010	0.20131
ivex.003	0.48514	0.33274	0.65544	0.01183	0.51486
lisp.000 (dup)	0.85269	0.25793	0.70906	0.03301	0.14731
lisp.001	0.85318	0.24862	0.71186	0.03952	0.14682
pasc.001	0.49258	0.42761	0.54628	0.02611	0.50742
spic.000 (dup)	0.55129	0.45503	0.52754	0.01743	0.44871
spic.001	0.59019	0.43677	0.54539	0.01784	0.40981
umil1	0.71593	0.16833	0.81256	0.01911	0.28407
umil2	0.81697	0.15012	0.84312	0.00676	0.18303
Mean	0.66840	0.29763	0.66598	0.03639	0.33160
Std Dev	0.12648	0.09142	0.09396	0.04303	0.12648
cc1	0.76295	0.22934	0.73371	0.03694	0.23705
spice	0.48403	0.38042	0.60562	0.01396	0.51597
tex	0.24994	0.55940	0.42384	0.01676	0.75006
Mean	0.49897	0.38972	0.58772	0.02255	0.50103
Std Dev	0.25683	0.16523	0.15571	0.01254	0.25683

Table A.17. State Transition Probabilities - Write References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.66896	0.15733	0.60855	0.23413	0.33104
boyer	0.49963	0.12213	0.30289	0.57498	0.50037
compile-rb	0.36934	0.56519	0.39585	0.03896	0.63066
compile-str	0.32992	0.47859	0.46204	0.05937	0.67008
fft	0.69853	0.08827	0.61508	0.29665	0.30147
glisp-comp	0.57404	0.26407	0.56732	0.16861	0.42596
glisp-pay	0.13269	0.48899	0.47044	0.04057	0.86731
qsim	0.48031	0.24195	0.42765	0.33040	0.51969
reducer	0.08036	0.86728	0.06994	0.06278	0.91964
tmycin	0.40256	0.33211	0.56894	0.09895	0.59744
Mean	0.42363	0.36059	0.44887	0.19054	0.57637
Std Dev	0.20603	0.24165	0.16679	0.17268	0.20603
dec0.000	0.45188	0.46114	0.50526	0.03360	0.54812
fora.000	0.49198	0.48772	0.45130	0.06097	0.50802
forf.003	0.45691	0.48466	0.44194	0.07340	0.54309
fsxzz.000	0.50035	0.64553	0.29141	0.06306	0.49965
ivex.000	0.41788	0.35686	0.56054	0.08260	0.58212
linp.000	0.43153	0.64482	0.29873	0.05644	0.56847
lisp.000	0.51215	0.30830	0.61414	0.07756	0.48785
macr.000	0.53433	0.38504	0.48786	0.12710	0.46567
memxx.000	0.85725	0.46843	0.35966	0.17191	0.14275
pasc.000	0.29479	0.51384	0.47281	0.01334	0.70521
savec.003	0.37161	0.60508	0.36166	0.03326	0.62839
spic.000	0.36321	0.64491	0.34121	0.01387	0.63679
ue02.000	0.36167	0.31923	0.60093	0.07984	0.63833
Mean	0.46504	0.48658	0.44519	0.06823	0.53496
Std Dev	0.13698	0.12182	0.10899	0.04404	0.13698
dec0.001	0.28405	0.61006	0.38613	0.00381	0.71595
dec1.001	0.25412	0.59560	0.39861	0.00580	0.74588
dia0	0.61992	0.24484	0.74164	0.01352	0.38008
forl.000	0.35357	0.54787	0.41790	0.03423	0.64643
forl.001	0.39664	0.56354	0.40683	0.02964	0.60336
ivex.000 (dup)	0.35743	0.39909	0.53504	0.06587	0.64257
ivex.003	0.40227	0.66461	0.32596	0.00943	0.59773
lisp.000 (dup)	0.50018	0.27950	0.61747	0.10303	0.49982
lisp.001	0.49158	0.25393	0.63399	0.11208	0.50842
pasc.001	0.33085	0.65337	0.27259	0.07404	0.66915
spic.000 (dup)	0.29007	0.70187	0.28623	0.01190	0.70993
spic.001	0.31552	0.74153	0.24815	0.01032	0.68448
umil1	0.58050	0.29432	0.67581	0.02988	0.41950
umil2	0.53172	0.28349	0.70354	0.01297	0.46828
Mean	0.40774	0.48812	0.47499	0.03689	0.59226
Std Dev	0.11721	0.18584	0.17204	0.03682	0.11721
cc1	0.30066	0.36027	0.60606	0.03367	0.69934
spice	0.52206	0.50959	0.45144	0.03897	0.47794
tex	0.24997	0.89974	0.00044	0.09982	0.75003
Mean	0.35756	0.58987	0.35265	0.05749	0.64244
Std Dev	0.14470	0.27855	0.31466	0.03676	0.14470

Appendix B. *ATUM Reference Preprocessing*

Three traces in the ATUM collection were common to both the DIN and MIT sets: ivex.000, lisp.000, and spic.000. These traces provide some insight into how the word (MIT) traces were preprocessed and the reliability of the two sets of traces. The MIT ATUM traces which were at the word level also contained PID identifiers which were used to denote context switching. In the DIN traces, these identifiers were left in the trace as read references with leading bits added to the reference. For example, the process id (PID) 143C1Eh showed up in the MIT spic.000 trace eleven times. Converting this address back to a byte address (assuming word alignment), this PID is 50F078h. This address shows up in the DIN spic.000 trace eleven times as well only as a read reference with the address 150F078h.

In addition, their appears to be some differences in the least significant bit of corresponding word references. For example, the byte address 136Ah in the lisp.000 DIN trace should show up as a word address of 4DAh in the MIT lisp.000, but instead is shown as 4DBh. The DIN trace also differs in some of the other bit positions as well. In the spic.000 DIN trace a reference is made to the addresses around 71FE7DEXh several times. This byte address translates to 1C7F9F78h when changed to a word. In the MIT trace, this address is 1FFF9F78h. Four consecutive bit positions in the DIN trace differ. Other addresses in both traces indicate that it was the DIN trace that was switched.

Several techniques were tried to process the DIN traces in a manner similar to the MIT traces. Other sources indicate that the MIT traces were preprocessed by collapsing successive instruction fetches from the same word into a single word reference [PRZYB,33]. Three techniques were tried to process the byte references of the DIN trace set into word references. All three techniques first converted the byte addresses to word addresses and then eliminating addresses. The first technique was to eliminate consecutive references to the same address while taking the type of reference into account. This technique did not appear to eliminate the number of references needed to coincide

with the corresponding MIT trace. In particular, the number of instruction and write references differed a great deal. The second technique was to simply eliminate consecutive references to the same address without regard to type. This technique also did not reduce the number of references to match. In particular, the instruction reference count was still too high. The third technique was to eliminate an instruction reference if the previous instruction reference referenced the same word. This technique brought the number of instruction references in line with that of the MIT instruction reference count. In addition, the probability of having the same stack distance, a characteristic which will be discussed later seemed to agree.

Although the number of references did not agree exactly, this technique was only an attempt in trying to preprocess the traces in the same manner as the MIT. One concern is that some instruction and write references are being eliminated when in fact they should remain as part of the trace. Taking this into account, results using the DIN trace set should be taken as less reliable. Results are annotated as to the granularity of the DIN trace used.

Appendix C. *Spatial Distance Probabilities*

Table C.1. Spatial Distance Probabilities (New-New/Old-New) - All References

TRACE NAME	NN/NO S Jmp	NN/NO SmJmp	NN/NO SBJmp	NN/NO MdJmp	NN/NO MBJmp	NN/NO BgJmp	NN/NO BBJmp
biaslisp	0.292	0.477	0.262	0.020	0.036	0.102	0.103
boyer	0.017	0.486	0.475	0.011	0.011	0.008	0.009
compile-rb	0.372	0.500	0.137	0.051	0.075	0.120	0.118
compile-str	0.366	0.498	0.162	0.048	0.070	0.112	0.110
fft	0.375	0.476	0.213	0.005	0.005	0.137	0.164
glisp-comp	0.319	0.494	0.176	0.029	0.035	0.123	0.143
glisp-pay	0.683	0.719	0.060	0.037	0.048	0.072	0.064
qsim	0.233	0.503	0.275	0.047	0.056	0.060	0.059
reducer	0.748	0.813	0.107	0.011	0.013	0.028	0.028
tmycin	0.316	[illegible]	0.271	0.016	[illegible]	0.078	0.079
Mean	0.372	0.550	0.214	0.027	0.037	0.084	0.088
Std Dev	0.210	0.117	0.117	0.017	0.025	0.043	0.049
dec0.000	0.030	0.474	0.096	0.079	0.073	0.141	0.137
fora.000	0.011	0.424	0.080	0.104	0.051	0.168	0.173
forf.003	0.011	0.439	0.070	0.078	0.060	0.174	0.179
fsxzz.000	0.009	0.309	0.057	0.252	0.246	0.069	0.067
ivex.000	0.008	0.603	0.049	0.051	0.043	0.120	0.134
linp.000	0.002	0.844	0.007	0.032	0.031	0.042	0.044
lisp.000	0.001	0.512	0.133	0.087	0.099	0.081	0.088
macr.000	0.007	0.442	0.027	0.121	0.053	0.166	0.191
memxx.000	0.007	0.735	0.061	0.034	0.026	0.074	0.070
pasc.000	0.020	0.474	0.104	0.049	0.031	0.169	0.173
savec.003	0.008	0.561	0.059	0.074	0.061	0.123	0.122
spic.000	0.046	0.560	0.033	0.076	0.087	0.122	0.122
ue02.000	0.017	0.479	0.051	0.087	0.066	0.145	0.172
Mean	0.014	0.527	0.064	0.086	0.071	0.123	0.129
Std Dev	0.012	0.139	0.034	0.056	0.057	0.044	0.049
dec0.001	0.369	0.479	0.092	0.106	0.120	0.102	0.101
dec1.001	0.378	0.494	0.084	[illegible]	0.109	0.107	0.107
dis0	0.351	0.517	0.054	0.094	0.095	0.119	0.121
forl.000	0.383	0.499	0.050	0.145	0.122	0.090	0.094
forl.001	0.374	0.568	0.081	0.067	0.057	0.115	0.112
ivex.000 (dup)	0.555	0.641	0.061	0.053	0.055	0.095	0.095
ivex.003	0.452	0.601	0.055	0.061	0.056	0.115	0.112
lisp.000 (dup)	0.485	0.527	0.165	0.096	0.105	0.053	0.054
lisp.001	0.440	0.482	0.193	0.101	0.115	0.054	0.055
pasc.001	0.511	0.524	0.007	0.264	0.176	0.015	0.014
spic.000 (dup)	0.373	0.554	0.042	0.091	0.105	0.104	0.104
spic.001	0.299	0.523	0.043	0.155	0.185	0.048	0.046
umil1	0.390	0.553	0.070	0.081	0.074	0.111	0.111
umil2	0.290	0.452	0.145	0.068	0.062	0.137	0.136
Mean	0.404	0.530	0.082	0.106	0.103	0.090	0.090
Std Dev	0.076	0.051	0.052	0.054	0.041	0.034	0.034
ccl	0.037	0.748	0.023	0.076	0.038	0.057	0.058
spice	0.152	0.776	0.009	0.046	0.031	0.068	0.070
tex	0.425	0.428	0.000	0.000	0.000	0.286	0.286
Mean	0.205	0.651	0.011	0.041	0.023	0.137	0.138
Std Dev	0.199	0.193	0.012	0.038	0.020	0.129	0.128

Table C.2. Spatial Distance Probabilities (New-New) - All References

TRACE NAME	NN S Jmp	NN SmJmp	NN SBJmp	NN MdJmp	NN MBJmp	NN BgJmp	NN BBJmp
biaslisp	0.054	0.062	0.878	0.004	0.007	0.031	0.018
boyer	0.010	0.012	0.941	0.018	0.018	0.001	0.010
compile-rb	0.377	0.493	0.198	0.045	0.095	0.083	0.086
compile-str	0.361	0.470	0.239	0.043	0.087	0.081	0.080
fft	0.174	0.176	0.664	0.015	0.015	0.004	0.126
glisp-comp	0.339	0.429	0.313	0.030	0.050	0.074	0.104
glisp-pay	0.812	0.835	0.042	0.037	0.042	0.021	0.023
qsim	0.290	0.361	0.465	0.026	0.029	0.054	0.065
reducer	0.843	0.864	0.101	0.009	0.009	0.009	0.008
tmycin	0.447	0.485	0.416	0.014	0.016	0.029	0.040
Mean	0.371	0.419	0.426	0.024	0.037	0.039	0.056
Std Dev	0.279	0.287	0.313	0.014	0.032	0.032	0.042
dec0.000	0.001	0.474	0.059	0.079	0.037	0.176	0.175
fora.000	0.001	0.460	0.060	0.065	0.034	0.224	0.157
forf.003	0.003	0.488	0.047	0.076	0.043	0.174	0.172
fsxzz.000	0.000	0.261	0.071	0.037	0.471	0.088	0.072
ivex.000	0.002	0.505	0.051	0.071	0.042	0.163	0.168
linp.000	0.007	0.409	0.036	0.063	0.056	0.210	0.226
lisp.000	0.001	0.720	0.028	0.038	0.037	0.088	0.089
macr.000	0.002	0.534	0.028	0.074	0.040	0.162	0.162
memxx.000	0.000	0.466	0.136	0.068	0.024	0.166	0.140
pasc.000	0.002	0.505	0.090	0.058	0.024	0.155	0.168
savec.003	0.001	0.513	0.069	0.071	0.019	0.177	0.151
spic.000	0.013	0.483	0.024	0.092	0.092	0.151	0.158
ue02.000	0.012	0.523	0.048	0.065	0.041	0.161	0.162
Mean	0.003	0.488	0.057	0.066	0.074	0.161	0.154
Std Dev	0.004	0.099	0.030	0.015	0.121	0.039	0.038
dec0.001	0.310	0.478	0.067	0.089	0.091	0.137	0.138
dec1.001	0.335	0.487	0.065	0.087	0.081	0.140	0.140
dia0	0.371	0.536	0.021	0.077	0.076	0.146	0.144
forl.000	0.377	0.542	0.049	0.085	0.080	0.120	0.124
forl.001	0.356	0.587	0.049	0.055	0.047	0.139	0.123
ivex.000 (dup)	0.374	0.522	0.063	0.064	0.057	0.146	0.148
ivex.003	0.313	0.451	0.068	0.073	0.064	0.169	0.175
lisp.000 (dup)	0.689	0.735	0.032	0.041	0.047	0.074	0.071
lisp.001	0.695	0.740	0.032	0.041	0.045	0.070	0.072
pasc.001	0.908	0.922	0.006	0.014	0.015	0.021	0.022
spic.000 (dup)	0.394	0.477	0.029	0.096	0.119	0.146	0.133
spic.001	0.181	0.223	0.042	0.285	0.217	0.127	0.106
umil1	0.376	0.531	0.031	0.078	0.078	0.145	0.137
umil2	0.148	0.240	0.131	0.157	0.141	0.184	0.147
Mean	0.416	0.534	0.049	0.089	0.083	0.126	0.120
Std Dev	0.208	0.183	0.030	0.065	0.050	0.043	0.040
cc1	0.000	0.838	0.000	0.044	0.012	0.053	0.053
spice	0.000	0.766	0.000	0.027	0.010	0.099	0.098
tex	0.000	0.570	0.000	0.054	0.027	0.174	0.175
Mean	0.000	0.725	0.000	0.042	0.016	0.109	0.109
Std Dev	0.000	0.139	0.000	0.014	0.009	0.061	0.062

Table C.3. Spatial Distance Probabilities (Old-New) - All References

TRACE NAME	ON S Jmp	ON SmJmp	ON SBJmp	ON MdJmp	ON MBJmp	ON BgJmp	ON BBJmp
biaslisp	0.391	0.650	0.005	0.027	0.049	0.132	0.137
boyer	0.023	0.948	0.020	0.003	0.004	0.015	0.010
compile-rb	0.377	0.493	0.198	0.045	0.095	0.083	0.086
compile-str	0.374	0.543	0.038	0.057	0.042	0.161	0.159
fft	0.468	0.613	0.006	0.000	0.001	0.199	0.181
glisp-comp	0.296	0.565	0.026	0.027	0.019	0.177	0.186
glisp-pay	0.297	0.370	0.112	0.037	0.066	0.224	0.191
qsim	0.165	0.671	0.051	0.071	0.089	0.068	0.050
reducer	0.146	0.495	0.142	0.020	0.033	0.149	0.161
tmycin	0.100	0.607	0.032	0.019	0.039	0.158	0.145
Mean	0.264	0.595	0.063	0.031	0.044	0.137	0.131
Std Dev	0.147	0.152	0.065	0.022	0.032	0.064	0.062
dec0.000	0.044	0.474	0.113	0.078	0.089	0.124	0.122
fora.000	0.016	0.405	0.090	0.125	0.061	0.138	0.181
forf.003	0.017	0.407	0.085	0.080	0.071	0.174	0.183
fsxzz.000	0.016	0.351	0.046	0.437	0.052	0.053	0.061
ivex.000	0.011	0.652	0.048	0.041	0.044	0.097	0.118
linp.000	0.001	0.920	0.002	0.027	0.027	0.013	0.011
lisp.000	0.000	0.394	0.193	0.116	0.133	0.076	0.088
macr.000	0.009	0.396	0.026	0.144	0.060	0.169	0.205
memxx.000	0.010	0.038	0.817	0.023	0.027	0.046	0.049
pasc.000	0.031	0.455	0.113	0.043	0.036	0.177	0.176
savec.003	0.014	0.605	0.049	0.078	0.100	0.074	0.094
spic.000	0.071	0.618	0.040	0.064	0.084	0.100	0.094
ue02.000	0.020	0.449	0.053	0.102	0.083	0.135	0.178
Mean	0.020	0.474	0.129	0.104	0.067	0.106	0.120
Std Dev	0.019	0.203	0.212	0.107	0.031	0.052	0.061
dec0.001	0.400	0.479	0.104	0.114	0.135	0.084	0.084
dec1.001	0.412	0.498	0.100	0.109	0.130	0.081	0.082
dia0	0.327	0.496	0.093	0.114	0.117	0.087	0.093
forl.000	0.385	0.476	0.050	0.176	0.145	0.075	0.078
forl.001	0.384	0.558	0.099	0.073	0.063	0.102	0.105
ivex.000 (dup)	0.636	0.695	0.060	0.048	0.054	0.072	0.071
ivex.003	0.562	0.719	0.045	0.051	0.049	0.072	0.064
lisp.000 (dup)	0.372	0.411	0.238	0.127	0.137	0.041	0.046
lisp.001	0.323	0.363	0.267	0.129	0.147	0.046	0.048
pasc.001	0.090	0.102	0.007	0.529	0.347	0.008	0.007
spic.000 (dup)	0.356	0.615	0.052	0.087	0.094	0.070	0.082
spic.001	0.343	0.635	0.043	0.107	0.173	0.018	0.024
umil1	0.401	0.570	0.099	0.083	0.071	0.085	0.092
umil2	0.340	0.526	0.150	0.037	0.035	0.120	0.132
Mean	0.381	0.510	0.101	0.127	0.121	0.069	0.072
Std Dev	0.122	0.155	0.074	0.121	0.078	0.031	0.033
ccl	0.075	0.656	0.046	0.108	0.065	0.062	0.063
spice	0.283	0.784	0.018	0.063	0.049	0.042	0.044
tex	0.426	0.427	0.000	0.000	0.000	0.286	0.287
Mean	0.261	0.622	0.021	0.057	0.038	0.130	0.131
Std Dev	0.177	0.181	0.023	0.054	0.034	0.135	0.135

Table C.4. Spatial Distance Probabilities (New-New/Old-New) - Inst References

TRACE NAME	NN/NO S Jmp	NN/NO SmJmp	NN/NO SBJmp	NN/NO MdJmp	NN/NO MBJmp	NN/NO BgJmp	NN/NO BBJmp
biaslisp	0.821	0.894	0.014	0.039	0.034	0.009	0.010
boyer	0.623	0.811	0.094	0.038	0.057	0.000	0.000
compile-rb	0.716	0.862	0.013	0.041	0.035	0.025	0.024
compile-str	0.715	0.860	0.012	0.043	0.036	0.024	0.025
fft	0.835	0.890	0.009	0.018	0.028	0.028	0.027
glisp-comp	0.702	0.823	0.017	0.051	0.049	0.029	0.031
glisp-pay	0.701	0.853	0.020	0.040	0.029	0.029	0.029
qsim	0.685	0.816	0.024	0.062	0.060	0.019	0.019
reducer	0.698	0.811	0.022	0.060	0.064	0.021	0.022
tmycin	0.705	0.847	0.013	0.040	0.034	0.034	0.032
Mean	0.720	0.847	0.024	0.043	0.043	0.022	0.022
Std Dev	0.063	0.031	0.025	0.012	0.014	0.010	0.010
dec0.000	0.001	0.849	0.004	0.069	0.037	0.021	0.020
fora.000	0.001	0.840	0.004	0.083	0.049	0.012	0.012
forf.003	0.002	0.842	0.003	0.078	0.047	0.015	0.015
fsxzz.000	0.000	0.861	0.005	0.066	0.039	0.014	0.015
ivex.000	0.002	0.845	0.003	0.067	0.044	0.021	0.020
linp.000	0.008	0.850	0.005	0.063	0.045	0.018	0.019
lisp.000	0.004	0.852	0.001	0.060	0.047	0.020	0.020
macr.000	0.001	0.854	0.003	0.071	0.046	0.014	0.012
memxx.000	0.000	0.855	0.006	0.070	0.041	0.015	0.013
pasc.000	0.001	0.831	0.002	0.090	0.042	0.018	0.017
savec.003	0.000	0.872	0.006	0.062	0.036	0.012	0.012
spic.000	0.017	0.851	0.002	0.062	0.037	0.023	0.025
ue02.000	0.001	0.854	0.003	0.065	0.044	0.018	0.016
Mean	0.003	0.850	0.004	0.070	0.043	0.017	0.017
Std Dev	0.005	0.010	0.002	0.009	0.004	0.004	0.004
dec0.001	0.595	0.865	0.017	0.038	0.037	0.022	0.021
dec1.001	0.609	0.855	0.016	0.048	0.047	0.016	0.018
dia0	0.571	0.852	0.013	0.059	0.053	0.012	0.011
forl.000	0.607	0.863	0.009	0.056	0.053	0.010	0.009
forl.001	0.592	0.871	0.010	0.059	0.053	0.003	0.004
ivex.000 (dup)	0.629	0.856	0.012	0.053	0.048	0.016	0.015
ivex.003	0.640	0.876	0.009	0.049	0.043	0.012	0.011
lisp.000 (dup)	0.628	0.868	0.017	0.032	0.040	0.023	0.020
lisp.001	0.627	0.871	0.014	0.029	0.035	0.028	0.023
pasc.001	0.622	0.857	0.015	0.044	0.046	0.019	0.019
spic.000 (dup)	0.580	0.862	0.009	0.053	0.046	0.014	0.016
spic.001	0.586	0.875	0.008	0.044	0.044	0.016	0.013
umil1	0.582	0.853	0.011	0.058	0.053	0.013	0.012
umil2	0.511	0.834	0.007	0.055	0.062	0.024	0.018
Mean	0.598	0.861	0.012	0.048	0.047	0.016	0.015
Std Dev	0.033	0.011	0.003	0.010	0.007	0.007	0.005
cc1	0.000	0.910	0.001	0.053	0.020	0.008	0.008
spice	0.000	0.933	0.001	0.037	0.016	0.007	0.006
tex	0.000	0.892	0.000	0.057	0.025	0.013	0.013
Mean	0.000	0.912	0.001	0.049	0.020	0.009	0.009
Std Dev	0.000	0.021	0.001	0.011	0.005	0.003	0.004

Table C.5. Spatial Distance Probabilities (New-New) - Inst References

TRACE NAME	NN S Jmp	NN SmJmp	NN SBJmp	NN MdJmp	NN MBJmp	NN BgJmp	NN BBJmp
biaslisp	0.870	0.942	0.003	0.017	0.027	0.003	0.008
boyer	0.707	0.927	0.000	0.000	0.073	0.000	0.000
compile-rb	0.773	0.926	0.006	0.029	0.014	0.012	0.013
compile-str	0.773	0.925	0.006	0.030	0.015	0.011	0.013
fft	0.874	0.922	0.010	0.010	0.019	0.029	0.010
glisp-comp	0.768	0.896	0.008	0.032	0.025	0.019	0.020
glisp-pay	0.762	0.920	0.013	0.028	0.013	0.012	0.014
qsim	0.742	0.887	0.009	0.052	0.030	0.011	0.011
reducer	0.800	0.918	0.012	0.025	0.019	0.009	0.017
tmycin	0.758	0.903	0.006	0.033	0.017	0.021	0.020
Mean	0.783	0.917	0.007	0.026	0.025	0.013	0.013
Std Dev	0.053	0.016	0.004	0.014	0.018	0.008	0.006
dec0.000	0.001	0.885	0.003	0.061	0.027	0.012	0.012
fora.000	0.001	0.880	0.003	0.070	0.032	0.008	0.007
forf.003	0.002	0.874	0.002	0.068	0.037	0.009	0.010
fsxzz.000	0.000	0.888	0.005	0.060	0.031	0.007	0.009
ivex.000	0.002	0.880	0.002	0.059	0.035	0.011	0.013
linp.000	0.008	0.868	0.003	0.059	0.042	0.014	0.014
lisp.000	0.005	0.886	0.000	0.052	0.038	0.011	0.013
macr.000	0.001	0.882	0.002	0.062	0.037	0.008	0.009
memxx.000	0.000	0.879	0.005	0.065	0.034	0.008	0.009
pasc.000	0.001	0.856	0.001	0.086	0.038	0.010	0.009
savec.003	0.089	0.891	0.006	0.057	0.030	0.008	0.008
spic.000	0.018	0.881	0.002	0.056	0.029	0.016	0.016
ue02.000	0.001	0.893	0.002	0.054	0.033	0.009	0.009
Mean	0.010	0.880	0.003	0.062	0.034	0.010	0.011
Std Dev	0.024	0.010	0.002	0.009	0.004	0.003	0.003
dec0.001	0.646	0.913	0.007	0.029	0.022	0.014	0.015
dec1.001	0.653	0.899	0.008	0.038	0.034	0.010	0.011
dia0	0.607	0.898	0.008	0.044	0.036	0.007	0.007
forl.000	0.642	0.904	0.005	0.043	0.037	0.006	0.005
forl.001	0.641	0.925	0.004	0.039	0.027	0.002	0.003
ivex.000 (dup)	0.671	0.907	0.007	0.038	0.032	0.008	0.008
ivex.003	0.667	0.911	0.006	0.038	0.030	0.009	0.006
lisp.000 (dup)	0.665	0.919	0.007	0.024	0.021	0.014	0.015
lisp.001	0.663	0.919	0.007	0.022	0.020	0.016	0.016
pasc.001	0.649	0.891	0.012	0.035	0.033	0.015	0.014
spic.000 (dup)	0.620	0.899	0.007	0.043	0.032	0.010	0.009
spic.001	0.620	0.899	0.003	0.041	0.037	0.010	0.010
umil1	0.616	0.900	0.008	0.044	0.035	0.008	0.005
umil2	0.570	0.913	0.002	0.035	0.030	0.013	0.007
Mean	0.638	0.907	0.007	0.037	0.030	0.010	0.009
Std Dev	0.028	0.010	0.002	0.007	0.006	0.004	0.004
cc1	0.000	0.937	0.000	0.043	0.010	0.005	0.005
spice	0.000	0.950	0.000	0.032	0.009	0.004	0.005
tex	0.000	0.903	0.000	0.058	0.026	0.006	0.007
Mean	0.000	0.930	0.000	0.044	0.015	0.005	0.006
Std Dev	0.000	0.024	0.000	0.013	0.010	0.001	0.001

Table C.6. Spatial Distance Probabilities (Old-New) - Inst References

TRACE NAME	ON S Jmp	ON SmJmp	ON SBJmp	ON MdJmp	ON MBJmp	ON BgJmp	ON BBJmp
biaslisp	0.208	0.292	0.146	0.312	0.125	0.083	0.042
boyer	0.333	0.417	0.417	0.166	0.000	0.000	0.000
compile-rb	0.270	0.364	0.066	0.138	0.199	0.125	0.108
compile-str	0.272	0.369	0.059	0.147	0.194	0.124	0.107
fft	0.167	0.333	0.000	0.167	0.167	0.000	0.333
glisp-comp	0.331	0.414	0.069	0.161	0.184	0.086	0.086
glisp-pay	0.137	0.233	0.082	0.151	0.178	0.192	0.164
qsim	0.381	0.442	0.106	0.115	0.218	0.058	0.061
reducer	0.277	0.373	0.063	0.203	0.247	0.070	0.044
tmycin	0.287	0.406	0.063	0.098	0.168	0.133	0.132
Mean	0.266	0.364	0.107	0.166	0.168	0.087	0.108
Std Dev	0.076	0.064	0.115	0.059	0.067	0.060	0.093
dec0.000	0.000	0.433	0.016	0.161	0.147	0.128	0.115
fora.000	0.000	0.401	0.018	0.229	0.238	0.057	0.057
forf.003	0.001	0.383	0.018	0.215	0.189	0.101	0.094
fsxzz.000	0.003	0.330	0.009	0.188	0.190	0.151	0.132
ivex.000	0.001	0.390	0.013	0.168	0.154	0.145	0.130
linp.000	0.000	0.384	0.055	0.178	0.123	0.123	0.137
lisp.000	0.000	0.345	0.017	0.190	0.190	0.155	0.103
macr.000	0.000	0.418	0.008	0.198	0.197	0.093	0.086
memxx.000	0.000	0.352	0.015	0.183	0.180	0.162	0.108
pasc.000	0.000	0.408	0.014	0.163	0.111	0.149	0.155
savec.003	0.000	0.422	0.021	0.167	0.172	0.115	0.103
spic.000	0.000	0.445	0.013	0.140	0.140	0.118	0.144
ue02.000	0.001	0.404	0.013	0.198	0.174	0.115	0.096
Mean	0.000	0.393	0.018	0.183	0.170	0.124	0.112
Std Dev	0.001	0.035	0.012	0.024	0.034	0.029	0.027
dec0.001	0.229	0.523	0.087	0.105	0.146	0.077	0.062
dec1.001	0.203	0.442	0.095	0.146	0.169	0.072	0.076
dia0	0.213	0.386	0.056	0.207	0.216	0.062	0.073
forl.000	0.194	0.394	0.048	0.209	0.231	0.057	0.061
forl.001	0.224	0.473	0.055	0.213	0.240	0.009	0.010
ivex.000 (dup)	0.237	0.368	0.054	0.195	0.206	0.086	0.091
ivex.003	0.238	0.355	0.055	0.219	0.234	0.059	0.078
lisp.000 (dup)	0.246	0.348	0.116	0.116	0.232	0.116	0.072
lisp.001	0.239	0.352	0.099	0.099	0.197	0.155	0.098
pasc.001	0.174	0.291	0.070	0.209	0.267	0.081	0.082
spic.000 (dup)	0.134	0.454	0.042	0.164	0.202	0.067	0.071
spic.001	0.120	0.540	0.080	0.080	0.140	0.100	0.060
umil1	0.232	0.365	0.047	0.208	0.247	0.066	0.067
umil2	0.058	0.231	0.038	0.212	0.308	0.115	0.096
Mean	0.196	0.394	0.067	0.170	0.217	0.080	0.071
Std Dev	0.056	0.085	0.024	0.051	0.046	0.034	0.021
cc1	0.000	0.420	0.012	0.242	0.202	0.064	0.060
spice	0.000	0.593	0.014	0.130	0.137	0.060	0.066
tex	0.000	0.500	0.000	0.000	0.000	0.250	0.250
Mean	0.000	0.504	0.009	0.124	0.113	0.125	0.125
Std Dev	0.000	0.087	0.008	0.121	0.103	0.10[illegible]	0.108

Table C.7. Spatial Distance Probabilities (New-New/Old-New) - Data References

TRACE NAME	NN/NO S Jmp	NN/NO SmJmp	NN/NO SBJmp	NN/NO MdJmp	NN/NO MBJmp	NN/NO BgJmp	NN/NO BBJmp
biaslisp	0.291	0.482	0.265	0.020	0.036	0.099	0.098
boyer	0.016	0.486	0.476	0.011	0.011	0.008	0.008
compile-rb	0.334	0.456	0.159	0.016	0.019	0.176	0.174
compile-str	0.330	0.459	0.195	0.017	0.020	0.155	0.154
fft	0.375	0.476	0.213	0.005	0.005	0.137	0.164
glisp-comp	0.292	0.475	0.170	0.019	0.018	0.145	0.173
glisp-pay	0.702	0.727	0.059	0.033	0.042	0.074	0.065
qsim	0.186	0.478	0.293	0.050	0.057	0.063	0.059
reducer	0.774	0.834	0.098	0.009	0.009	0.025	0.025
tmycin	0.304	0.524	0.278	0.013	0.020	0.083	0.082
Mean	0.360	0.540	0.221	0.019	0.024	0.097	0.100
Std Dev	0.224	0.131	0.117	0.013	0.016	0.056	0.062
dec0.000	0.050	0.365	0.168	0.093	0.102	0.139	0.133
fora.000	0.019	0.258	0.153	0.136	0.064	0.191	0.198
forf.003	0.022	0.270	0.155	0.088	0.075	0.203	0.209
fsxzz.000	0.013	0.142	0.093	0.337	0.340	0.046	0.042
ivex.000	0.015	0.574	0.100	0.041	0.040	0.110	0.135
linp.000	0.001	0.896	0.016	0.027	0.028	0.018	0.015
lisp.000	0.000	0.502	0.167	0.088	0.099	0.068	0.076
macr.000	0.014	0.278	0.062	0.159	0.054	0.202	0.245
memxx.000	0.011	0.747	0.092	0.027	0.028	0.054	0.052
pasc.000	0.032	0.396	0.179	0.040	0.030	0.176	0.179
savec.003	0.023	0.288	0.198	0.132	0.141	0.122	0.119
spic.000	0.065	0.544	0.065	0.094	0.120	0.090	0.087
ue02.000	0.034	0.338	0.116	0.109	0.084	0.156	0.197
Mean	0.023	0.431	0.120	0.105	0.093	0.121	0.130
Std Dev	0.019	0.214	0.054	0.082	0.083	0.063	0.072
dec0.001	0.305	0.364	0.181	0.140	0.159	0.080	0.076
dec1.001	0.281	0.344	0.208	0.120	0.140	0.097	0.091
dia0	0.082	0.187	0.229	0.144	0.156	0.141	0.143
forl.000	0.319	0.381	0.100	0.182	0.143	0.094	0.100
forl.001	0.210	0.367	0.198	0.079	0.063	0.147	0.146
ivex.000 (dup)	0.596	0.635	0.116	0.043	0.048	0.078	0.080
ivex.003	0.443	0.579	0.141	0.059	0.045	0.088	0.088
lisp.000 (dup)	0.496	0.523	0.198	0.093	0.099	0.043	0.044
lisp.001	0.443	0.471	0.225	0.102	0.112	0.044	0.046
pasc.001	0.516	0.519	0.010	0.278	0.181	0.006	0.006
spic.000 (dup)	0.345	0.541	0.082	0.118	0.137	0.062	0.060
spic.001	0.287	0.534	0.055	0.174	0.200	0.019	0.018
umil1	0.250	0.370	0.197	0.078	0.071	0.144	0.140
umil2	0.280	0.438	0.198	0.037	0.042	0.143	0.142
Mean	0.347	0.447	0.153	0.118	0.114	0.085	0.084
Std Dev	0.137	0.119	0.069	0.064	0.053	0.047	0.047
ccl	0.138	0.457	0.095	0.167	0.112	0.086	0.083
spice	0.370	0.717	0.039	0.086	0.080	0.039	0.039
tex	0.426	0.427	0.000	0.000	0.000	0.287	0.286
Mean	0.311	0.534	0.045	0.084	0.064	0.137	0.136
Std Dev	0.153	0.159	0.048	0.084	0.058	0.132	0.132

Table C.8. Spatial Distance Probabilities (New-New) - Data References

TRACE NAME	NN S Jmp	NN SmJmp	NN SBJmp	NN MdJmp	NN MBJmp	NN BgJmp	NN BBJmp
biaslisp	0.046	0.049	0.913	0.003	0.002	0.016	0.017
boyer	0.009	0.944	0.010	0.019	0.018	0.001	0.008
compile-rb	0.398	0.455	0.242	0.012	0.017	0.126	0.148
compile-str	0.370	0.421	0.306	0.013	0.015	0.114	0.131
fft	0.166	0.166	0.642	0.015	0.014	0.002	0.161
glisp-comp	0.324	0.373	0.346	0.020	0.027	0.100	0.134
glisp-pay	0.850	0.857	0.037	0.035	0.034	0.020	0.017
qsim	0.246	0.276	0.543	0.027	0.031	0.058	0.065
reducer	0.871	0.876	0.090	0.008	0.006	0.007	0.013
tmycin	0.448	0.462	0.440	0.015	0.012	0.031	0.040
Mean	0.373	0.488	0.357	0.017	0.018	0.048	0.073
Std Dev	0.295	0.308	0.287	0.009	0.010	0.049	0.063
dec0.000	0.031	0.399	0.224	0.067	0.079	0.106	0.125
fora.000	0.006	0.325	0.272	0.057	0.053	0.203	0.090
forf.003	0.037	0.333	0.220	0.099	0.073	0.139	0.136
fsxzz.000	0.004	0.092	0.132	0.026	0.688	0.030	0.032
ivex.000	0.009	0.373	0.249	0.076	0.071	0.122	0.109
linp.000	0.002	0.274	0.150	0.187	0.192	0.103	0.094
lisp.000	0.001	0.724	0.120	0.029	0.021	0.048	0.058
macr.000	0.010	0.389	0.169	0.086	0.101	0.116	0.139
memxx.000	0.006	0.348	0.349	0.057	0.054	0.100	0.092
pasc.000	0.019	0.470	0.232	0.050	0.051	0.094	0.103
savec.003	0.003	0.332	0.303	0.093	0.094	0.085	0.093
spic.000	0.003	0.488	0.087	0.128	0.151	0.068	0.078
ue02.000	0.031	0.462	0.179	0.076	0.074	0.121	0.088
Mean	0.012	0.385	0.207	0.079	0.131	0.103	0.095
Std Dev	0.013	0.144	0.076	0.043	0.173	0.043	0.030
dec0.001	0.190	0.216	0.208	0.125	0.270	0.099	0.082
dec1.001	0.226	0.270	0.259	0.106	0.177	0.108	0.080
dia0	0.113	0.245	0.314	0.127	0.103	0.095	0.116
forl.000	0.346	0.443	0.235	0.062	0.061	0.111	0.088
forl.001	0.210	0.298	0.359	0.055	0.067	0.138	0.083
ivex.000 (dup)	0.301	0.384	0.304	0.079	0.056	0.096	0.081
ivex.003	0.194	0.248	0.320	0.116	0.067	0.138	0.111
lisp.000 (dup)	0.698	0.744	0.123	0.017	0.032	0.043	0.041
lisp.001	0.678	0.737	0.135	0.016	0.032	0.044	0.036
pasc.001	0.964	0.967	0.012	0.006	0.004	0.006	0.005
spic.000 (dup)	0.428	0.521	0.092	0.125	0.167	0.048	0.047
spic.001	0.300	0.348	0.068	0.307	0.216	0.031	0.030
umil1	0.083	0.194	0.366	0.084	0.089	0.119	0.148
umil2	0.007	0.038	0.362	0.102	0.123	0.195	0.180
Mean	0.338	0.404	0.226	0.095	0.105	0.091	0.081
Std Dev	0.270	0.256	0.120	0.074	0.077	0.051	0.048
cc1	0.028	0.464	0.134	0.254	0.105	0.024	0.019
spice	0.053	0.715	0.077	0.069	0.077	0.032	0.030
tex	0.000	0.000	0.001	0.000	0.000	0.428	0.571
Mean	0.027	0.393	0.071	0.108	0.061	0.161	0.207
Std Dev	0.027	0.363	0.067	0.131	0.054	0.231	0.316

Table C.9. Spatial Distance Probabilities (Old-New) - Data References

TRACE NAME	ON S Jmp	ON SmJmp	ON SBJmp	ON MdJmp	ON MBJmp	ON BgJmp	ON BBJmp
biaslisp	0.390	0.655	0.005	0.027	0.049	0.132	0.132
boyer	0.022	0.950	0.020	0.003	0.004	0.015	0.008
compile-rb	0.256	0.457	0.060	0.020	0.023	0.236	0.204
compile-str	0.281	0.506	0.057	0.022	0.026	0.207	0.182
fft	0.477	0.626	0.006	0.000	0.001	0.203	0.164
glisp-comp	0.265	0.564	0.018	0.019	0.010	0.184	0.205
glisp-pay	0.280	0.353	0.123	0.028	0.067	0.229	0.200
qsim	0.120	0.700	0.018	0.075	0.086	0.068	0.053
reducer	0.068	0.523	0.159	0.013	0.030	0.154	0.121
tmycin	0.083	0.618	0.028	0.009	0.031	0.164	0.150
Mean	0.224	0.595	0.049	0.022	0.033	0.159	0.142
Std Dev	0.148	0.161	0.052	0.021	0.027	0.071	0.066
dec0.000	0.061	0.347	0.137	0.107	0.114	0.157	0.138
fora.000	0.025	0.226	0.095	0.174	0.069	0.185	0.251
forf.003	0.013	0.231	0.114	0.081	0.076	0.243	0.255
fsxzz.000	0.020	0.185	0.059	0.597	0.047	0.059	0.053
ivex.000	0.017	0.638	0.053	0.030	0.030	0.106	0.143
linp.000	0.001	0.951	0.004	0.013	0.013	0.011	0.008
lisp.000	0.000	0.384	0.191	0.118	0.140	0.079	0.088
macr.000	0.015	0.246	0.032	0.179	0.040	0.226	0.277
memxx.000	0.011	0.822	0.044	0.021	0.024	0.046	0.043
pasc.000	0.038	0.367	0.158	0.036	0.022	0.208	0.209
savec.003	0.039	0.253	0.112	0.164	0.179	0.152	0.140
spic.000	0.098	0.575	0.053	0.076	0.104	0.101	0.091
ue02.000	0.036	0.247	0.070	0.133	0.091	0.182	0.277
Mean	0.029	0.421	0.086	0.133	0.073	0.135	0.152
Std Dev	0.027	0.249	0.054	0.151	0.051	0.074	0.094
dec0.001	0.367	0.444	0.166	0.148	0.099	0.070	0.073
dec1.001	0.317	0.391	0.176	0.129	0.116	0.089	0.099
dia0	0.063	0.153	0.178	0.155	0.188	0.168	0.158
forl.000	0.312	0.365	0.064	0.214	0.165	0.089	0.103
forl.001	0.210	0.397	0.128	0.089	0.062	0.151	0.173
ivex.000 (dup)	0.678	0.705	0.064	0.033	0.045	0.073	0.080
ivex.003	0.539	0.708	0.072	0.036	0.036	0.068	0.080
lisp.000 (dup)	0.387	0.402	0.238	0.134	0.135	0.043	0.048
lisp.001	0.332	0.345	0.267	0.142	0.150	0.044	0.052
pasc.001	0.059	0.062	0.009	0.555	0.362	0.006	0.006
spic.000 (dup)	0.292	0.554	0.075	0.113	0.118	0.071	0.069
spic.001	0.282	0.605	0.051	0.122	0.195	0.014	0.013
umil1	0.307	0.431	0.139	0.075	0.064	0.153	0.138
umil2	0.334	0.517	0.166	0.024	0.026	0.133	0.134
Mean	0.320	0.434	0.128	0.141	0.126	0.084	0.088
Std Dev	0.160	0.183	0.075	0.131	0.088	0.051	0.050
ccl	0.233	0.450	0.061	0.093	0.118	0.139	0.139
spice	0.554	0.718	0.017	0.097	0.081	0.043	0.044
tex	0.685	0.685	0.000	0.000	0.000	0.201	0.114
Mean	0.491	0.618	0.026	0.063	0.066	0.128	0.099
Std Dev	0.233	0.146	0.031	0.055	0.060	0.080	0.049

Table C.10. Spatial Distance Probabilities (New-New/Old-New) - Read References

TRACE NAME	NN/NO S Jmp	NN/NO SmJmp	NN/NO SBJmp	NN/NO MdJmp	NN/NO MBJmp	NN/NO BgJmp	NN/NO BBJmp
biaslisp	0.384	0.490	0.055	0.065	0.041	0.174	0.175
boyer	0.230	0.353	0.385	0.108	0.120	0.017	0.017
compile-rb	0.359	0.463	0.106	0.019	0.024	0.195	0.193
compile-str	0.390	0.488	0.104	0.027	0.031	0.175	0.175
fft	0.281	0.449	0.036	0.067	0.067	0.170	0.211
glisp-comp	0.361	0.476	0.137	0.031	0.027	0.151	0.178
glisp-pay	0.321	0.422	0.129	0.063	0.096	0.155	0.135
qsim	0.375	0.523	0.157	0.088	0.095	0.067	0.070
reducer	0.839	0.867	0.044	0.020	0.023	0.022	0.024
tmycin	0.566	0.647	0.101	0.017	0.028	0.102	0.105
Mean	0.411	0.518	0.125	0.051	0.055	0.123	0.128
Std Dev	0.174	0.144	0.100	0.032	0.036	0.066	0.071
dec0.000	0.050	0.365	0.168	0.093	0.102	0.139	0.133
fora.000	0.019	0.258	0.153	0.136	0.064	0.191	0.198
forf.003	0.022	0.270	0.155	0.088	0.075	0.203	0.209
fsxzz.000	0.013	0.142	0.093	0.337	0.340	0.046	0.042
ivex.000	0.015	0.574	0.100	0.041	0.040	0.110	0.135
linp.000	0.001	0.896	0.016	0.027	0.028	0.018	0.015
lisp.000	0.000	0.502	0.167	0.088	0.099	0.068	0.076
macr.000	0.014	0.278	0.062	0.159	0.054	0.202	0.245
memxx.000	0.011	0.747	0.092	0.027	0.028	0.054	0.052
pasc.000	0.032	0.396	0.179	0.040	0.030	0.176	0.179
savec.003	0.023	0.288	0.198	0.132	0.141	0.122	0.119
spic.000	0.065	0.544	0.065	0.094	0.120	0.090	0.087
ue02.000	0.034	0.338	0.116	0.109	0.084	0.156	0.197
Mean	0.023	0.431	0.120	0.105	0.093	0.121	0.130
Std Dev	0.019	0.214	0.054	0.082	0.083	0.063	0.072
dec0.001	0.305	0.364	0.181	0.140	0.159	0.080	0.076
dec1.001	0.281	0.344	0.208	0.120	0.140	0.097	0.091
dia0	0.082	0.187	0.229	0.144	0.156	0.141	0.143
forl.000	0.319	0.381	0.100	0.162	0.143	0.094	0.100
forl.001	0.210	0.367	0.198	0.079	0.063	0.147	0.146
ivex.000 (dup)	0.596	0.635	0.116	0.043	0.048	0.078	0.080
ivex.003	0.443	0.579	0.141	0.059	0.045	0.088	0.088
lisp.000 (dup)	0.496	0.523	0.198	0.093	0.099	0.043	0.044
lisp.001	0.443	0.471	0.225	0.102	0.112	0.044	0.046
pasc.001	0.516	0.519	0.010	0.278	0.181	0.006	0.006
spic.000 (dup)	0.345	0.541	0.082	0.118	0.137	0.062	0.060
spic.001	0.287	0.534	0.055	0.174	0.200	0.019	0.018
umil1	0.250	0.370	0.197	0.078	0.071	0.144	0.140
umil2	0.280	0.438	0.198	0.037	0.042	0.143	0.142
Mean	0.347	0.447	0.153	0.118	0.114	0.085	0.084
Std Dev	0.137	0.119	0.069	0.064	0.053	0.047	0.047
cc1	0.182	0.362	0.106	0.132	0.133	0.136	0.131
spice	0.196	0.669	0.068	0.099	0.082	0.042	0.040
tex	0.997	0.998	0.001	0.000	0.000	0.001	0.000
Mean	0.458	0.676	0.058	0.077	0.072	0.060	0.057
Std Dev	0.467	0.318	0.053	0.069	0.067	0.069	0.067

Table C.11. Spatial Distance Probabilities (New-New) - Read References

TRACE NAME	NN S Jmp	NN SmJmp	NN SBJmp	NN MdJmp	NN MBJmp	NN BgJmp	NN BBJmp
biaslisp	0.752	0.800	0.057	0.003	0.009	0.021	0.110
boyer	0.374	0.378	0.318	0.110	0.111	0.009	0.074
compile-rb	0.455	0.517	0.125	0.015	0.019	0.158	0.166
compile-str	0.527	0.582	0.103	0.018	0.018	0.139	0.140
fft	0.678	0.679	0.002	0.014	0.014	0 002	0.289
glisp-comp	0.444	0.500	0.171	0.025	0.032	0.118	0.154
glisp-pay	0.402	0.485	0.158	0.114	0.116	0.069	0.058
qsim	0.456	0.495	0.250	0.042	0.045	0.080	0.088
reducer	0.476	0.542	0.179	0.074	0.058	0.071	0.076
tmycin	0.660	0.688	0.101	0.032	0.027	0.067	0.085
Mean	0.522	0.567	0.146	0.045	0.045	0.073	0.124
Std Dev	0.129	0.123	0.091	0.040	0.039	0.053	0.069
dec0.000	0.031	0.399	0.224	0.067	0.079	0.106	0.125
fora.000	0.006	0.325	0.272	0.057	0.053	0.203	0.090
forf.003	0.037	0.333	0.220	0.099	0.073	0.139	0.136
fsxzz.000	0.004	0.092	0.132	0.026	0.688	0.030	0.032
ivex.000	0.009	0.373	0.249	0.076	0.071	0.122	0.109
linp.000	0.002	0.274	0.150	0.187	0.192	0.103	0.094
lisp.000	0.001	0.724	0.120	0.029	0.021	0.048	0.058
macr.000	0.010	0.389	0.169	0.086	0.101	0.116	0.139
memxx.000	0.000	0.348	0.349	0.057	0.054	0.100	0.092
pasc.000	0.019	0.470	0.232	0.050	0.051	0.094	0.103
savec.003	0.003	0.332	0.303	0.093	0.094	0.085	0.093
spic.000	0.003	0.488	0.087	0.128	0.151	0.068	0.078
ue02.000	0.031	0.462	0.179	0.076	0.074	0.121	0.088
Mean	0.012	0.385	0.207	0.079	0.131	0.103	0.095
Std Dev	0.013	0.144	0.076	0.043	0.173	0.043	0.030
dec0.001	0.190	0.216	0.208	0.125	0.270	0.099	0.082
dec1.001	0.226	0.270	0.259	0.106	0.177	0.108	0.080
dia0	0.113	0.245	0.314	0.127	0.103	0.095	0.116
forl.000	0.346	0.443	0.235	0.062	0.061	0.111	0.088
forl.001	0.210	0.298	0.359	0.055	0.067	0.138	0.083
ivex.000 (dup)	0.301	0.384	0 304	0.079	0.056	0.096	0.081
ivex.003	0.194	0.248	0.320	0.116	0.067	0.138	0.111
lisp.000 (dup)	0.698	0.744	0.123	0.017	0.032	0.043	0.041
lisp.001	0.678	0.737	0.135	0.016	0.032	0.044	0.036
pasc.001	0.964	0.967	0.012	0.006	0.004	0.006	0.005
spic.000 (dup)	0.428	0.521	0.092	0.125	0.167	0.048	0.047
spic.001	0.300	0.348	0.068	0.307	0.216	0.031	0.030
umil1	0.083	0.194	0.366	0.084	0.089	0.119	0.148
umil2	0.007	0.038	0.362	0.102	0.123	0.195	0.180
Mean	0.338	0.404	0.225	0.095	0.105	0.091	0.081
Std Dev	0.270	0.256	0.120	0.074	0.077	0.051	0.048
cc1	0.101	0.441	0.177	0.097	0.080	0.112	0.093
spice	0.133	0.803	0.085	0.030	0.026	0.022	0.034
tex	0.997	0.998	0.001	0.000	0.000	0.001	0.000
Mean	0.410	0.747	0.088	0.042	0.035	0.045	0.042
Std Dev	0.508	0.283	0.088	0.050	0.041	0.059	0.047

Table C.12. Spatial Distance Probabilities (Old-New) - Read References

TRACE NAME	ON S Jmp	ON SmJmp	ON SBJmp	ON MdJmp	ON MBJmp	ON BgJmp	ON BBJmp
biaslisp	0.217	0.349	0.054	0.094	0.055	0.244	0.204
boyer	0.213	0.350	0.393	0.107	0.121	0.018	0.011
compile-rb	0.254	0.404	0.086	0.023	0.028	0.235	0.224
compile-str	0.226	0.375	0.106	0.039	0.048	0.217	0.215
fft	0.070	0.327	0.054	0.095	0.095	0.258	0.171
glisp-comp	0.294	0.457	0.110	0.036	0.023	0.178	0.196
glisp-pay	0.260	0.375	0.107	0.026	0.080	0.219	0.193
qsim	0.314	0.544	0.087	0.123	0.133	0.058	0.055
reducer	0.882	0.906	0.028	0.014	0.019	0.016	0.017
tmycin	0.511	0.624	0.102	0.009	0.029	0.122	0.114
Mean	0.324	0.471	0.113	0.057	0.063	0.157	0.140
Std Dev	0.225	0.180	0.102	0.043	0.042	0.095	0.084
dec0.000	0.061	0.347	0.137	0.107	0.114	0.157	0.138
fora.000	0.025	0.226	0.095	0.174	0.069	0.185	0.251
forf.003	0.013	0.231	0.114	0.081	0.076	0.243	0.255
fsxzz.000	0.020	0.185	0.059	0.597	0.047	0.059	0.053
ivex.000	0.017	0.638	0.053	0.030	0.030	0.106	0.143
linp.000	0.001	0.951	0.004	0.013	0.013	0.011	0.008
lisp.000	0.000	0.384	0.191	0.118	0.140	0.079	0.088
macr.000	0.015	0.246	0.032	0.179	0.040	0.226	0.277
memxx.000	0.011	0.822	0.044	0.021	0.024	0.046	0.043
pasc.000	0.038	0.367	0.158	0.036	0.022	0.208	0.209
savec.003	0.039	0.253	0.112	0.164	0.179	0.152	0.140
spic.000	0.098	0.575	0.053	0.076	0.104	0.101	0.091
ue02.000	0.036	0.247	0.070	0.133	0.091	0.182	0.277
Mean	0.029	0.421	0.086	0.133	0.073	0.135	0.152
Std Dev	0.027	0.249	0.054	0.151	0.051	0.074	0.094
dec0.001	0.367	0.444	0.166	0.148	0.099	0.070	0.073
dec1.001	0.317	0.391	0.176	0.129	0.116	0.089	0.099
dia0	0.063	0.153	0.178	0.155	0.188	0.168	0.158
forl.000	0.312	0.365	0.064	0.214	0.165	0.089	0.103
forl.001	0.210	0.397	0.128	0.089	0.062	0.151	0.173
ivex.000 (dup)	0.678	0.705	0.064	0.033	0.045	0.073	0.080
ivex.003	0.539	0.708	0.072	0.036	0.036	0.068	0.080
lisp.000 (dup)	0.387	0.402	0.238	0.134	0.135	0.043	0.048
lisp.001	0.332	0.345	0.267	0.142	0.150	0.044	0.052
pasc.001	0.059	0.062	0.009	0.555	0.362	0.006	0.006
spic.000 (dup)	0.292	0.554	0.075	0.113	0.118	0.071	0.069
spic.001	0.282	0.605	0.051	0.122	0.195	0.014	0.013
umil1	0.307	0.431	0.139	0.075	0.064	0.153	0.138
umil2	0.334	0.517	0.166	0.024	0.026	0.133	0.134
Mean	0.320	0.434	0.128	0.141	0.126	0.084	0.088
Std Dev	0.160	0.183	0.075	0.131	0.088	0.051	0.050
cc1	0.207	0.338	0.084	0.143	0.149	0.144	0.142
spice	0.264	0.527	0.049	0.174	0.141	0.064	0.045
tex	0.997	0.997	0.000	0.000	0.000	0.002	0.001
Mean	0.489	0.621	0.044	0.106	0.097	0.070	0.063
Std Dev	0.441	0.339	0.042	0.093	0.084	0.071	0.072

Table C.13. Spatial Distance Probabilities (New-New/Old-New) - Write References

TRACE NAME	S Jmp	SmJmp	SBJmp	MdJmp	MBJmp	BgJmp	BBJmp
biaslisp	0.304	0.514	0.294	0.016	0.005	0.086	0.085
boyer	0.000	0.499	0.499	0.000	0.000	0.001	0.001
compile-rb	0.426	0.590	0.276	0.005	0.003	0.061	0.065
compile-str	0.371	0.552	0.335	0.007	0.005	0.050	0.051
fft	0.364	0.509	0.240	0.000	0.000	0.095	0.156
glisp-comp	0.284	0.561	0.249	0.013	0.002	0.077	0.098
glisp-pay	0.830	0.852	0.043	0.003	0.002	0.055	0.045
qsim	0.171	0.568	0.384	0.003	0.003	0.024	0.018
reducer	0.837	0.901	0.074	0.000	0.000	0.013	0.012
tmycin	0.279	0.581	0.370	0.001	0.001	0.023	0.024
Mean	0.387	0.613	0.276	0.005	0.002	0.049	0.055
Std Dev	0.264	0.143	0.138	0.006	0.002	0.032	0.047
dec0.000	0.066	0.437	0.207	0.065	0.066	0.118	0.107
fora.000	0.015	0.291	0.194	0.144	0.079	0.136	0.156
forf.003	0.011	0.232	0.164	0.072	0.061	0.283	0.188
fsxzz.000	0.038	0.339	0.368	0.076	0.076	0.076	0.065
ivex.000	0.024	0.299	0.196	0.061	0.052	0.215	0.177
linp.000	0.003	0.073	0.068	0.382	0.397	0.042	0.038
lisp.000	0.000	0.757	0.032	0.019	0.011	0.091	0.090
macr.000	0.017	0.292	0.060	0.133	0.083	0.190	0.242
memxx.000	0.011	0.819	0.098	0.014	0.018	0.028	0.023
pasc.000	0.012	0.405	0.339	0.046	0.049	0.080	0.081
savec.003	0.026	0.182	0.363	0.104	0.136	0.107	0.108
spic.000	0.029	0.566	0.117	0.061	0.094	0.084	0.078
ue02.000	0.031	0.392	0.148	0.093	0.082	0.144	0.141
Mean	0.022	0.391	0.181	0.098	0.093	0.123	0.115
Std Dev	0.017	0.214	0.114	0.094	0.097	0.072	0.063
dec0.001	0.367	0.403	0.348	0.045	0.048	0.082	0.074
dec1.001	0.315	0.362	0.402	0.038	0.041	0.078	0.079
dia0	0.024	0.085	0.585	0.076	0.047	0.104	0.103
forl.000	0.320	0.387	0.140	0.137	0.109	0.094	0.133
forl.001	0.260	0.456	0.257	0.056	0.048	0.090	0.093
ivex.000 (dup)	0.319	0.397	0.263	0.081	0.064	0.105	0.090
ivex.003	0.378	0.554	0.272	0.028	0.030	0.062	0.054
lisp.000 (dup)	0.771	0.798	0.033	0.010	0.012	0.076	0.071
lisp.001	0.763	0.799	0.038	0.011	0.011	0.073	0.068
pasc.001	0.525	0.526	0.007	0.268	0.195	0.002	0.002
spic.000 (dup)	0.513	0.601	0.138	0.080	0.106	0.040	0.035
spic.001	0.446	0.553	0.083	0.201	0.141	0.011	0.011
umil1	0.285	0.346	0.369	0.061	0.039	0.090	0.095
umil2	0.748	0.764	0.079	0.045	0.030	0.042	0.040
Mean	0.431	0.502	0.215	0.081	0.066	0.068	0.068
Std Dev	0.215	0.199	0.170	0.074	0.053	0.033	0.036
cc1	0.079	0.528	0.121	0.190	0.093	0.034	0.034
spice	0.438	0.809	0.038	0.066	0.059	0.014	0.014
tex	0.750	0.750	0.000	0.000	0.000	0.125	0.125
Mean	0.422	0.696	0.053	0.085	0.051	0.058	0.058
Std Dev	0.336	0.148	0.062	0.096	0.047	0.059	0.059

Table C.14. Spatial Distance Probabilities (New-New) - Write References

TRACE NAME	S Jmp	SmJmp	SBJmp	MdJmp	MBJmp	BgJmp	BBJmp
biaslisp	0.022	0.032	0.887	0.006	0.006	0.066	0.003
boyer	0.000	0.000	0.998	0.001	0.001	0.000	0.000
compile-rb	0.518	0.522	0.436	0.000	0.000	0.013	0.029
compile-str	0.456	0.460	0.496	0.001	0.001	0.013	0.029
fft	0.003	0.003	0.796	0.001	0.000	0.199	0.001
glisp-comp	0.370	0.390	0.581	0.002	0.000	0.012	0.015
glisp-pay	0.932	0.932	0.049	0.003	0.001	0.013	0.002
qsim	0.238	0.238	0.738	0.000	0.006	0.006	0.012
reducer	0.908	0.909	0.080	0.000	0.000	0.001	0.010
tmycin	0.365	0.368	0.619	0.000	0.001	0.006	0.006
Mean	0.381	0.385	0.568	0.001	0.002	0.033	0.011
Std Dev	0.341	0.340	0.316	0.002	0.002	0.061	0.011
dec0.000	0.026	0.541	0.279	0.036	0.021	0.067	0.056
fora.000	0.008	0.430	0.282	0.124	0.042	0.083	0.039
forf.003	0.018	0.318	0.217	0.042	0.046	0.275	0.102
fsxzz.000	0.046	0.185	0.590	0.089	0.059	0.045	0.032
ivex.000	0.026	0.445	0.270	0.052	0.035	0.136	0.062
linp.000	0.003	0.115	0.105	0.024	0.691	0.036	0.029
lisp.000	0.001	0.851	0.040	0.032	0.002	0.033	0.042
ma[illegible].000	0.028	0.461	0.093	0.247	0.034	0.140	0.025
me[illegible]x.000	0.053	0.250	0.569	0.063	0.040	0.043	0.035
pa[illegible]000	0.014	0.425	0.449	0.031	0.032	0.038	0.025
savec.003	0.028	0.248	0.448	0.104	0.089	0.060	0.051
spic.000	0.019	0.739	0.103	0.042	0.043	0.029	0.044
ue02.000	0.038	0.556	0.192	0.108	0.047	0.062	0.035
Mean	0.024	0.428	0.280	0.076	0.091	0.081	0.044
Std Dev	0.016	0.212	0.183	0.061	0.181	0.069	0.021
dec0.001	0.460	0.483	0.426	0.018	0.008	0.034	0.031
dec1.001	0.400	0.438	0.476	0.012	0.009	0.031	0.034
dia0	0.038	0.084	0.799	0.025	0.023	0.031	0.038
forl.000	0.465	0.520	0.177	0.183	0.078	0.020	0.022
forl.001	0.397	0.535	0.308	0.038	0.031	0.032	0.056
ivex.000 (dup)	0.445	0.514	0.320	0.065	0.041	0.037	0.023
ivex.003	0.457	0.499	0.397	0.016	0.014	0.042	0.032
lisp.000 (dup)	0.839	0.881	0.060	0.014	0.008	0.008	0.029
lisp.001	0.815	0.869	0.059	0.015	0.008	0.008	0.041
pasc.001	0.742	0.743	0.009	0.182	0.064	0.001	0.001
spic.000 (dup)	0.654	0.737	0.110	0.073	0.052	0.014	0.014
spic.001	0.642	0.674	0.054	0.127	0.127	0.010	0.008
umil1	0.292	0.319	0.554	0.027	0.030	0.030	0.040
umil2	0.703	0.710	0.116	0.045	0.039	0.032	0.058
Mean	0.525	0.572	0.276	0.060	0.038	0.024	0.030
Std Dev	0.221	0.217	0.233	0.061	0.034	0.013	0.016
cc1	0.014	0.503	0.141	0.236	0.103	0.009	0.008
spice	0.106	0.808	0.063	0.073	0.043	0.005	0.008
tex	0.999	0.999	0.000	0.000	0.000	0.000	0.001
Mean	0.373	0.770	0.068	0.103	0.049	0.005	0.006
Std Dev	0.544	0.250	0.071	0.121	0.052	0.005	0.004

Table C.15. Spatial Distance Probabilities (Old-New) - Write References

TRACE NAME	S Jmp	SmJmp	SBJmp	MdJmp	MBJmp	BgJmp	BBJmp
biaslisp	0.443	0.752	0.000	0.021	0.004	0.096	0.127
boyer	0.000	0.999	0.000	0.000	0.000	0.001	0.000
compile-rb	0.268	0.706	0.004	0.015	0.007	0.145	0.123
compile-str	0.198	0.740	0.008	0.019	0.014	0.123	0.096
fft	0.520	0.727	0.000	0.000	0.000	0.051	0.222
glisp-comp	0.220	0.688	0.002	0.020	0.003	0.125	0.162
glisp-pay	0.164	0.328	0.001	0.003	0.006	0.328	0.334
qsim	0.099	0.925	0.001	0.007	0.001	0.043	0.023
reducer	0.015	0.808	0.002	0.001	0.001	0.150	0.038
tmycin	0.151	0.896	0.001	0.001	0.003	0.049	0.050
Mean	0.208	0.757	0.002	0.009	0.004	0.111	0.117
Std Dev	0.168	0.183	0.002	0.009	0.004	0.091	0.102
dec0.000	0.116	0.310	0.121	0.101	0.120	0.180	0.168
fora.000	0.022	0.147	0.104	0.166	0.117	0.190	0.276
forf.003	0.004	0.130	0.101	0.108	0.079	0.293	0.289
fsxzz.000	0.029	0.492	0.146	0.063	0.092	0.107	0.100
ivex.000	0.022	0.096	0.093	0.075	0.076	0.324	0.336
linp.000	0.002	0.019	0.021	0.855	0.009	0.051	0.045
lisp.000	0.000	0.667	0.024	0.007	0.019	0.146	0.137
macr.000	0.008	0.145	0.032	0.034	0.126	0.234	0.429
memxx.000	0.003	0.914	0.020	0.006	0.014	0.025	0.021
pasc.000	0.006	0.358	0.076	0.082	0.090	0.181	0.213
savec.003	0.021	0.069	0.218	0.105	0.215	0.186	0.207
spic.000	0.047	0.262	0.142	0.093	0.184	0.181	0.138
ue02.000	0.019	0.102	0.070	0.068	0.145	0.289	0.326
Mean	0.023	0.285	0.090	0.136	0.099	0.184	0.207
Std Dev	0.031	0.265	0.059	0.220	0.063	0.089	0.121
dec0.001	0.132	0.202	0.152	0.115	0.148	0.202	0.181
dec1.001	0.065	0.141	0.184	0.113	0.136	0.218	0.208
dia0	0.016	0.086	0.453	0.108	0.062	0.148	0.143
forl.000	0.056	0.143	0.074	0.053	0.166	0.229	0.335
forl.001	0.052	0.335	0.181	0.084	0.074	0.180	0.146
ivex.000 (dup)	0.092	0.187	0.161	0.110	0.105	0.227	0.210
ivex.003	0.260	0.635	0.086	0.047	0.053	0.092	0.087
lisp.000 (dup)	0.703	0.716	0.005	0.005	0.016	0.145	0.113
lisp.001	0.708	0.728	0.016	0.007	0.015	0.140	0.094
pasc.001	0.086	0.088	0.003	0.443	0.461	0.003	0.002
spic.000 (dup)	0.167	0.266	0.207	0.098	0.237	0.105	0.087
spic.001	0.020	0.289	0.146	0.360	0.173	0.012	0.020
umil1	0.280	0.366	0.235	0.086	0.045	0.133	0.135
umil2	0.789	0.811	0.046	0.046	0.023	0.051	0.023
Mean	0.245	0.357	0.139	0.120	0.122	0.135	0.127
Std Dev	0.277	0.256	0.119	0.126	0.119	0.075	0.089
cc1	0.229	0.587	0.072	0.082	0.068	0.093	0.098
spice	0.743	0.810	0.014	0.058	0.073	0.022	0.023
tex	0.000	0.000	0.000	0.000	0.000	0.500	0.500
Mean	0.324	0.466	0.029	0.047	0.047	0.205	0.207
Std Dev	0.381	0.418	0.038	0.042	0.041	0.258	0.257

Appendix D. *Dispersion Data*

Table D.1. Dispersion for Blocksize of 4 - All References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	8344	0.049	3.804	0.646	0.269	2.926	1.456
boyer	5311	0.045	3.819	0.642	0.170	3.321	1.150
compile-rb	6575	0.336	2.658	1.258	0.526	1.894	1.312
compile-str	7318	0.316	2.737	1.263	0.508	1.969	1.339
fft	9898	0.016	3.935	0.394	0.304	2.783	1.584
glisp-comp	5170	0.297	2.813	1.286	0.502	1.992	1.338
glisp-pay	2597	0.206	3.177	1.200	0.328	2.689	1.382
qsim	3785	0.245	3.019	1.220	0.453	2.187	1.365
reducer	5205	0.101	3.597	0.920	0.187	3.253	1.186
tmycin	3605	0.214	3.146	1.190	0.410	2.358	1.348
Mean		0.183	3.270	1.002	0.366	2.537	1.346
Std Dev		0.121	0.482	0.328	0.133	0.532	0.123
dec0.001	2412	0.375	2.500	1.169	0.525	1.898	1.223
dec1.001	3694	0.371	2.517	1.158	0.522	1.913	1.195
dia0	5269	0.410	2.358	1.090	0.555	1.779	1.090
forl.000	6776	0.403	2.389	1.203	0.550	1.799	1.179
forl.001	5929	0.326	2.695	1.174	0.516	1.937	1.225
ivex.000 (dup)	10824	0.272	2.912	1.155	0.409	2.363	1.342
ivex.003	3098	0.340	2.640	1.187	0.490	2.039	1.235
lisp.000 (dup)	1782	0.203	3.186	1.166	0.413	2.347	1.403
lisp.001	2111	0.191	3.237	1.142	0.429	2.284	1.414
pasc.001	5434	0.114	3.544	0.893	0.486	2.055	1.591
spic.000 (dup)	2827	0.318	2.727	1.086	0.528	1.890	1.289
spic.001	2191	0.363	2.549	1.048	0.581	1.675	0.227
umil1	4710	0.389	2.445	1.101	0.531	1.875	1.156
umil2	926	0.297	2.411	1.149	0.579	1.679	1.266
Mean		0.312	2.722	1.123	0.508	1.967	1.203
Std Dev		0.089	0.366	0.079	0.057	0.227	0.308

Table D.2. Dispersion for Blocksize of 8 - All References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	4328	0.083	7.335	1.737	0.441	7.335	1.737
boyer	2743	0.076	7.394	1.729	0.158	6.738	2.261
compile-rb	4078	0.464	4.285	2.668	0.687	2.505	2.678
compile-str	4491	0.443	4.459	2.727	0.669	2.648	2.796
fft	5211	0.066	7.474	1.457	0.485	4.121	4.396
glisp-comp	3195	0.431	4.552	2.886	0.678	2.575	2.811
glisp-pay	1482	0.304	5.567	2.808	0.498	4.014	3.272
qsim	2215	0.355	5.158	2.789	0.576	3.395	3.009
reducer	2773	0.156	6.751	2.319	0.267	5.860	2.950
tmycin	2049	0.308	5.534	2.780	0.554	3.571	2.100
Mean		0.269	5.851	2.390	0.501	4.276	2.801
Std Dev		0.160	1.281	0.544	0.176	1.763	0.727
dec0.001	1571	0.520	3.838	2.480	0.689	2.486	2.395
dec1.001	2389	0.514	3.892	2.441	0.689	2.487	2.341
dia0	3351	0.537	3.708	2.193	0.713	2.295	2.112
forl.000	4408	0.541	3.673	2.526	0.714	2.291	2.343
forl.001	3648	0.452	4.380	2.601	0.680	2.558	2.516
ivex.000 (dup)	6544	0.398	4.816	2.649	0.565	3.483	2.963
ivex.003	8178	0.472	4.222	2.565	0.653	2.780	2.520
lisp.000 (dup)	1049	0.323	5.413	2.824	0.620	3.038	3.252
lisp.001	1232	0.307	5.546	2.808	0.645	2.836	3.326
pasc.001	3052	0.211	6.309	2.429	0.723	2.217	4.227
spic.000 (dup)	1756	0.451	4.391	2.490	0.687	2.502	2.618
spic.001	1310	0.467	0.426	2.475	0.709	2.330	2.402
umil1	2923	0.508	3.940	2.252	0.680	2.558	2.314
umil2	567	0.508	3.938	2.357	0.690	2.483	2.536
Mean		0.443	4.178	2.506	0.676	2.596	2.705
Std Dev		0.099	1.339	0.180	0.043	0.342	0.564

Table D.3. Dispersion for Blocksize of 16 - All References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	2278	0.129	13.935	4.423	0.587	6.612	9.461
boyer	1415	0.104	14.334	4.106	0.174	13.223	5.230
compile-rb	2495	0.562	7.004	5.223	0.801	3.192	5.071
compile-str	2732	0.542	7.331	5.416	0.790	3.367	5.357
fft	2624	0.072	14.842	3.107	0.618	6.110	10.532
glisp-comp	2034	0.553	7.151	5.851	0.810	3.041	5.140
glisp-pay	853	0.396	9.672	6.140	0.643	5.719	6.967
qsim	1314	0.457	8.696	5.915	0.684	5.049	6.303
reducer	1507	0.224	12.423	5.522	0.360	10.243	6.796
tmycin	1157	0.387	9.801	5.950	0.673	5.238	6.791
Mean		0.343	10.519	5.165	0.614	6.179	6.765
Std Dev		0.194	3.108	0.983	0.203	3.255	1.873
dec0.001	1062	0.645	5.678	4.630	0.811	3.030	4.163
dec1.001	1585	0.633	5.866	4.583	0.813	2.994	4.111
dia0	2158	0.640	5.758	4.135	0.827	2.771	3.785
forl.000	2942	0.656	5.503	4.792	0.833	2.665	4.118
forl.001	2267	0.559	7.049	5.249	0.807	3.091	4.776
ivex.000 (dup)	4089	0.518	7.708	5.494	0.697	4.852	5.892
ivex.003	1260	0.594	6.490	5.024	0.777	3.573	4.604
lisp.000 (dup)	650	0.454	8.735	6.233	0.786	3.425	6.437
lisp.001	754	0.434	9.062	6.222	0.814	2.981	6.551
pasc.001	1701	0.292	11.320	5.145	0.857	2.290	9.136
spic.000 (dup)	1109	0.565	6.952	5.019	0.814	2.977	4.920
spic.001	812	0.570	6.877	5.291	0.866	2.147	5.021
umil1	1831	0.607	6.289	4.040	0.796	3.259	4.300
umil2	373	0.626	5.987	4.769	0.805	3.121	4.279
Mean		0.557	7.091	5.045	0.807	3.084	5.150
Std Dev		0.102	1.639	0.649	0.040	0.640	1.442

Table D.4. Dispersion for Blocksize of 4 - Inst References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	194	0.164	3.345	1.105	0.231	3.077	1.102
boyer	20	0.325	2.700	1.145	0.463	2.150	0.964
compile-rb	2404	0.302	2.790	1.149	0.367	2.533	1.121
compile-str	2519	0.298	2.807	1.153	0.367	2.531	1.123
fft	32	0.141	3.438	0.899	0.203	3.188	0.982
glisp-comp	1239	0.287	2.851	1.161	0.378	2.490	1.120
glisp-pay	292	0.351	2.596	1.188	0.399	2.404	1.145
qsim	736	0.332	2.673	1.204	0.410	2.359	1.136
reducer	516	0.258	2.969	1.117	0.361	2.554	1.145
tmycin	457	0.306	2.777	1.145	0.371	2.514	1.122
Mean		0.276	2.895	1.127	0.355	2.580	1.096
Std Dev		0.070	0.282	0.085	0.079	0.316	0.066
dec0.001	974	0.332	2.670	0.994	0.392	2.430	0.994
dec1.001	1859	0.341	2.637	1.039	0.400	2.400	1.034
dia0	3450	0.374	2.503	0.014	0.428	2.290	1.003
forl.000	2670	0.359	2.565	1.040	0.409	2.365	1.021
forl.001	3298	0.327	2.691	1.046	0.402	2.392	1.031
ivex.000 (dup)	4613	0.326	2.697	1.047	0.389	2.443	1.053
ivex.003	1550	0.348	2.608	1.071	0.396	2.417	1.044
lisp.000 (dup)	298	0.342	2.631	1.073	0.392	2.433	1.031
lisp.001	317	0.341	2.634	1.062	0.392	2.432	1.018
pasc.001	596	0.358	2.569	1.009	0.404	2.383	1.028
spic.000 (dup)	1136	0.374	2.504	0.992	0.416	2.335	0.971
spic.001	301	0.375	2.502	0.984	0.412	2.352	0.974
umil1	2731	0.378	2.487	1.026	0.428	2.290	1.010
umil2	209	0.458	2.167	1.005	0.492	2.033	0.950
Mean	1714	0.360	2.562	0.957	0.411	2.357	1.012
Std Dev	1422	0.034	0.135	0.273	0.027	0.106	0.030

Table D.5. Dispersion for Blocksize of 8 - Inst References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	116	0.301	5.595	2.675	0.412	4.707	2.655
boyer	13	0.481	4.154	2.070	0.644	2.846	1.920
compile-rb	1514	0.446	4.431	2.355	0.544	3.650	2.212
compile-str	1586	0.443	4.458	2.383	0.545	3.644	2.223
fft	19	0.276	5.789	2.353	0.428	4.579	2.560
glisp-comp	790	0.441	4.471	2.412	0.565	3.482	2.191
glisp-pay	196	0.517	3.867	2.335	0.580	3.362	2.182
qsim	485	0.493	4.056	2.317	0.593	3.258	2.129
reducer	335	0.428	4.573	2.410	0.565	3.484	2.271
tmycin	294	0.460	4.316	2.270	0.557	3.544	2.129
Mean		0.429	4.571	2.358	0.543	3.656	2.247
Std Dev		0.079	0.630	0.149	0.071	0.571	0.213
dec0.001	578	0.438	4.500	2.112	0.526	3.789	2.085
dec1.001	1125	0.455	4.358	2.158	0.544	3.651	2.101
dia0	2092	0.484	4.128	2.024	0.569	3.447	1.985
forl.000	1602	0.466	4.275	2.120	0.547	3.624	2.063
forl.001	1937	0.427	4.581	2.185	0.546	3.628	2.159
ivex.000 (dup)	2747	0.434	4.530	2.158	0.529	3.767	2.169
ivex.003	959	0.473	4.216	2.199	0.547	3.621	2.095
lisp.000 (dup)	177	0.446	4.429	2.198	0.534	3.729	2.189
lisp.001	191	0.454	4.372	2.200	0.537	3.702	2.175
pasc.001	368	0.480	4.160	2.131	0.540	3.682	2.093
spic.000 (dup)	695	0.488	4.094	2.099	0.554	3.568	1.993
spic.001	180	0.477	4.183	2.136	0.532	3.744	2.086
umil1	1672	0.492	4.063	2.047	0.573	3.413	1.957
umil2	146	0.612	3.103	1.978	0.654	2.767	1.803
Mean		0.473	4.214	2.125	0.552	3.581	2.068
Std Dev		0.045	0.361	0.068	0.032	0.259	0.105

Table D.6. Dispersion for Blocksize of 16 - Inst References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	75	0.459	8.653	5.410	0.598	6.427	5.107
boyer	8	0.578	6.750	3.700	0.742	4.125	3.808
compile-rb	954	0.561	7.031	4.405	0.686	5.030	4.049
compile-str	1003	0.559	7.049	4.461	0.688	4.998	4.067
fft	12	0.426	9.167	4.981	0.609	6.250	4.868
glisp-comp	512	0.569	6.898	4.426	0.721	4.471	3.910
glisp-pay	133	0.644	5.699	4.045	0.725	4.406	3.548
qsim	312	0.606	6.304	3.768	0.726	4.378	3.492
reducer	220	0.565	6.964	4.284	0.718	4.518	3.867
tmycin	186	0.574	6.823	3.906	0.694	4.898	3.747
Mean		0.554	7.134	4.339	0.691	4.950	4.046
Std Dev		0.065	1.029	0.537	0.049	0.789	0.533
dec0.001	349	0.534	7.453	4.040	0.648	5.636	3.855
dec1.001	701	0.563	6.994	4.109	0.677	5.161	3.814
dia0	1263	0.573	6.838	3.837	0.689	4.971	3.702
forl.000	975	0.561	7.025	3.941	0.677	5.174	3.820
forl.001	1143	0.515	7.764	4.238	0.672	5.248	4.270
ivex.000 (dup)	1646	0.528	7.560	4.098	0.659	5.464	4.139
ivex.003	599	0.578	6.750	4.054	0.670	5.285	3.765
lisp.000 (dup)	111	0.559	7.063	4.271	0.677	5.162	3.980
lisp.001	120	0.565	6.958	4.210	0.682	5.083	3.932
pasc.001	231	0.586	6.628	3.938	0.675	5.203	3.822
spic.000 (dup)	434	0.590	6.555	3.980	0.683	5.071	3.729
spic.001	114	0.587	6.605	4.021	0.658	5.465	3.876
umil1	1020	0.584	6.660	3.901	0.697	4.850	3.658
umil2	104	0.728	4.356	3.101	0.774	3.615	2.741
Mean		0.575	6.801	3.981	0.681	5.099	3.793
Std Dev		0.050	0.796	0.284	0.030	0.474	0.345

Table D.7. Dispersion for Blocksize of 4 - Data References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	8167	0.048	3.807	0.645	0.260	2.958	1.435
boyer	5302	0.046	3.815	0.649	0.170	3.322	1.149
compile-rb	4484	0.399	2.402	1.319	0.537	1.851	1.243
compile-str	5127	0.368	2.528	1.336	0.509	1.966	1.286
fft	9866	0.016	3.936	0.390	0.303	2.787	1.582
glisp-comp	4085	0.326	2.696	1.340	0.497	2.013	1.283
glisp-pay	2336	0.198	3.208	1.211	0.310	2.759	1.381
qsim	3188	0.258	2.674	1.271	0.447	2.212	1.336
reducer	4781	0.101	3.595	0.948	0.161	3.356	1.114
tmycin	3231	0.221	3.118	1.226	0.400	2.400	1.315
Mean		0.198	3.178	1.034	0.359	2.562	1.312
Std Dev		0.141	0.584	0.352	0.139	0.554	0.136
dec0.001	1462	0.410	2.359	1.258	0.549	1.806	1.266
dec1.001	1872	0.410	2.362	1.253	0.550	1.801	1.245
dia0	1891	0.496	2.016	1.157	0.660	1.361	0.987
forl.000	4153	0.436	2.257	1.284	0.585	1.658	1.213
forl.001	2681	0.498	4.015	2.983	0.753	1.977	2.568
ivex.000 (dup)	6300	0.241	3.036	1.223	0.360	2.559	1.430
ivex.003	1577	0.339	2.642	1.299	0.480	2.080	1.333
lisp.000 (dup)	1488	0.178	3.290	1.159	0.391	2.438	1.430
lisp.001	1799	0.166	3.335	1.131	0.413	2.348	1.443
pasc.001	4889	0.089	3.643	0.825	0.483	2.066	1.656
spic.000 (dup)	1714	0.287	2.851	1.133	0.531	1.875	1.430
spic.001	1894	0.362	2.553	1.060	0.586	1.657	1.241
umil1	2033	0.418	2.329	1.197	0.587	1.652	1.243
umil2	721	0.383	2.469	1.180	0.572	1.712	1.337
Mean		0.337	2.797	1.296	0.536	1.928	1.416
Std Dev		0.127	0.587	0.500	0.106	0.340	0.366

Table D.8. Dispersion for Blocksize of 8 - Data References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	4234	0.082	7.344	1.735	0.432	4.544	3.993
boyer	2742	0.078	7.377	1.758	0.158	6.736	2.264
compile-rb	2943	0.543	3.660	2.737	0.697	2.423	2.471
compile-str	3300	0.509	3.928	2.845	0.671	2.633	2.640
fft	5193	0.065	7.479	1.451	0.484	4.129	4.394
glisp-comp	2609	0.472	4.221	2.991	0.678	2.576	2.664
glisp-pay	1334	0.298	5.617	2.883	0.482	4.145	3.302
qsim	1909	0.381	4.955	2.953	0.567	2.463	2.962
reducer	2558	0.160	6.720	2.429	0.237	6.104	2.828
tmycin	1858	0.322	5.421	2.899	0.547	3.624	3.042
Mean		0.291	5.672	2.468	0.495	3.938	3.056
Std Dev		0.186	1.485	0.592	0.182	1.531	0.674
dec0.001	1015	0.575	3.398	2.576	0.713	2.980	2.389
dec1.001	1302	0.576	3.396	2.577	0.716	2.273	2.359
dia0	1355	0.648	2.813	2.220	0.801	1.593	1.739
forl.000	3865	0.591	3.272	2.654	0.748	2.015	2.347
forl.001	1775	0.498	4.015	2.983	0.753	1.977	2.568
ivex.000 (dup)	3895	0.386	4.911	2.963	0.515	3.876	3.247
ivex.003	1010	0.484	4.126	2.885	0.636	2.911	2.724
lisp.000 (dup)	876	0.302	5.588	2.907	3.153	0.606	3.370
lisp.001	1046	0.283	5.735	2.868	0.638	2.893	3.467
pasc.001	2725	0.183	6.535	2.369	0.730	2.158	4.482
spic.000 (dup)	1092	0.441	4.474	2.728	0.692	2.464	2.821
spic.001	1133	0.467	4.267	2.525	0.707	2.347	2.389
umil1	1315	0.550	3.601	2.478	0.719	2.252	2.486
umil2	427	0.479	4.169	2.415	0.662	2.703	2.720
Mean		0.462	4.307	2.653	0.870	2.361	2.793
Std Dev		0.133	1.061	0.242	0.661	0.754	0.669

Table D.9. Dispersion for Blocksize of 16 - Data References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	2230	0.129	13.944	4.452	0.579	6.737	9.403
boyer	1415	0.106	14.296	4.162	0.174	13.213	5.245
compile-rb	1932	0.652	5.575	5.173	0.809	3.051	4.383
compile-str	2139	0.621	6.060	5.506	0.793	3.306	4.776
fft	2613	0.071	14.863	3.083	0.617	6.127	10.538
glisp-comp	1744	0.605	6.315	5.944	0.818	2.911	4.653
glisp-pay	768	0.390	9.757	6.384	0.630	5.919	7.094
qsim	1183	0.500	7.997	6.281	0.686	5.016	6.169
reducer	1405	0.235	12.235	5.858	0.335	10.637	6.675
tmycin	1071	0.412	9.405	6.236	0.672	5.247	6.654
Mean		0.372	10.045	5.308	0.611	6.216	6.559
Std Dev		0.224	3.586	1.092	0.209	3.344	2.045
dec0.001	735	0.707	4.693	4.587	0.829	2.731	3.910
dec1.001	924	0.701	4.785	4.659	0.831	2.702	3.897
dia0	1000	0.762	3.812	3.852	0.888	1.792	2.648
forl.000	2031	0.712	4 616	4.928	0.858	2.274	3.968
forl.001	1192	0.626	5.978	5.938	0.868	2.118	4.429
ivex.000 (dup)	2545	0.530	7.516	6.255	0.653	5.545	6.450
ivex.003	690	0.623	6.039	5.707	0.767	3.722	4.826
lisp.000 (dup)	543	0.437	9.015	6.525	0.778	3.552	6.695
lisp.001	639	0.413	9.388	6.475	0.812	3.014	6.839
pasc.001	1504	0.260	11.841	5.072	0.865	2.165	9.749
spic.000 (dup)	705	0.567	6.930	5.586	0.816	2.946	5.173
spic.001	701	0.569	6.897	5.462	0.867	2.123	5.065
umil1	880	0.664	5.381	4.838	0.822	2.844	4.332
umil2	276	0.597	6.449	5.135	0.777	3.572	4.673
Mean		0.583	6.667	5.358	0.817	2.936	5.190
Std Dev		0.138	2.200	0.778	0.060	0.960	1.762

Table D.10. Dispersion for Blocksize of 4 - Read References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	8159	0.066	3.735	0.686	0.591	1.635	2.263
boyer	4694	0.169	3.324	1.054	0.711	1.158	2.219
compile-rb	4465	0.418	2.327	1.281	0.628	1.486	1.251
compile-str	5098	0.393	2.428	1.296	0.605	1.581	1.301
fft	9865	0.068	3.729	0.706	0.619	1.525	2.368
glisp-comp	4067	0.371	2.516	1.280	0.639	1.446	1.329
glisp-pay	1568	0.369	2.522	1.270	0.618	1.530	1.363
qsim	3159	0.298	2.808	1.266	0.625	1.500	1.485
reducer	4747	0.109	3.565	0.968	0.711	1.155	2.454
tmycin	2989	0.323	2.707	1.251	0.624	1.503	1.437
Mean		0.258	2.966	1.106	0.637	1.452	1.747
Std Dev		0.141	0.563	0.242	0.041	0.164	0.506
dec0.001	1387	0.441	2.236	1.242	0.702	1.191	1.190
dec1.001	1782	0.449	2.205	1.227	0.688	1.246	1.144
dia0	1827	0.515	1.939	1.108	0.712	1.151	0.915
forl.000	3414	0.461	2.158	1.236	0.656	1.377	1.157
forl.001	2408	0.415	2.342	1.267	0.699	1.206	1.271
ivex.000 (dup)	6019	0.257	2.973	1.237	0.689	1.242	1.840
ivex.003	1378	0.408	2.368	1.320	0.608	1.566	1.271
lisp.000 (dup)	1317	0.299	2.804	1.235	0.720	1.118	1.739
lisp.001	1571	0.305	2.779	1.239	0.718	1.129	1.711
pasc.001	3514	0.051	3.797	0.707	0.515	1.942	1.876
spic.000 (dup)	1552	0.328	2.688	1.095	0.582	1.671	1.304
spic.001	1650	0.326	2.695	0.965	0.555	1.781	1.208
umil1	1933	0.437	2.251	1.163	0.717	1.133	1.179
umil2	647	0.421	2.315	1.131	0.736	1.057	1.279
Mean		0.365	2.539	1.155	0.664	1.344	1.363
Std Dev		0.117	0.469	0.157	0.070	0.281	0.299

Table D.11. Dispersion for Blocksize of 8 - Read References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	4229	0.099	7.206	1.746	0.776	1.793	5.539
boyer	2639	0.261	5.913	2.443	0.840	1.278	4.682
compile-rb	2938	0.558	3.536	2.614	0.773	1.813	2.423
compile-str	3293	0.530	3.759	2.698	0.754	1.969	2.578
fft	5192	0.114	7.085	1.716	0.758	1.937	5.293
glisp-comp	2604	0.509	3.930	2.754	0.791	1.669	2.619
glisp-pay	1007	0.509	3.928	2.496	0.784	1.731	2.451
qsim	1899	0.416	4.670	2.833	0.787	1.702	3.104
reducer	2549	0.170	6.639	2.450	0.836	1.312	5.394
tmycin	1849	0.453	4.375	2.588	0.786	1.711	2.855
Mean		0.362	5.104	2.434	0.788	1.691	3.694
Std Dev		0.182	1.457	0.392	0.029	0.231	1.352
dec0.001	977	0.603	3.175	2.512	0.837	1.301	2.068
dec1.001	1259	0.610	3.122	2.462	0.828	1.376	1.991
dia0	1329	0.667	2.665	2.080	0.844	1.251	1.586
forl.000	2339	0.606	3.149	2.526	0.807	1.544	2.133
forl.001	1649	0.573	3.420	2.637	0.837	1.301	2.278
ivex.000 (dup)	3729	0.400	4.798	2.953	0.823	1.419	3.542
ivex.003	922	0.558	3.539	2.709	0.760	1.923	2.393
lisp.000 (dup)	840	0.450	4.396	2.684	0.847	1.226	3.238
lisp.001	1002	0.455	4.357	2.682	0.845	1.237	3.201
pasc.001	1833	0.090	7.278	1.971	0.758	1.938	5.354
spic.000 (dup)	992	0.474	4.206	2.602	0.764	1.887	2.590
spic.001	926	0.400	4.801	2.277	0.732	2.144	2.884
umil1	1269	0.571	3.429	2.387	0.843	1.259	2.257
umil2	393	0.524	3.812	2.258	0.855	1.163	2.676
Mean		0.499	4.011	2.481	0.813	1.498	2.728
Std Dev		0.144	1.151	0.266	0.041	0.330	0.931

Table D.12. Dispersion for Blocksize of 16 - Read References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	2227	0.145	13.683	4.400	0.887	1.815	11.925
boyer	1414	0.310	11.036	4.909	0.910	1.443	9.630
compile-rb	1930	0.664	5.383	4.892	0.862	2.216	4.311
compile-str	2136	0.638	5.794	5.155	0.850	2.407	4.681
fft	2612	0.120	14.083	3.556	0.864	2.174	11.986
glisp-comp	1742	0.633	5.875	5.395	0.885	1.842	4.488
glisp-pay	663	0.627	5.965	4.673	0.879	1.929	4.262
qsim	1180	0.530	7.516	5.932	0.887	1.814	5.840
reducer	1403	0.246	12.062	5.833	0.903	1.559	10.623
tmycin	1069	0.527	7.568	4.837	0.877	1.970	5.776
Mean		0.444	8.897	4.958	0.880	1.917	7.352
Std Dev		0.216	3.460	0.693	0.018	0.294	3.286
dec0.001	709	0.727	4.375	4.502	0.913	1.386	3.225
dec1.001	901	0.727	4.362	4.388	0.908	1.474	3.132
dia0	987	0.776	3.589	3.555	0.913	1.393	2.441
forl.000	1642	0.720	4.486	4.692	0.900	1.604	3.363
forl.001	1105	0.681	5.103	5.045	0.913	1.388	3.862
ivex.000 (dup)	2424	0.539	7.382	6.207	0.901	1.582	6.001
ivex.003	645	0.684	5.059	5.059	0.867	2.129	3.857
lisp.000 (dup)	531	0.565	6.955	5.468	0.920	1.281	5.722
lisp.001	623	0.562	7.008	5.422	0.920	1.287	5.776
pasc.001	994	0.161	13.422	5.249	0.880	1.920	11.515
spic.000 (dup)	651	0.599	6.409	5.305	0.878	1.951	4.710
spic.001	529	0.475	8.405	5.100	0.862	2.212	6.334
umil1	856	0.682	5.084	4.598	0.910	1.440	3.784
umil2	259	0.639	5.784	4.703	0.921	1.263	4.550
Mean		0.610	6.244	4.949	0.900	1.594	4.877
Std Dev		0.155	2.481	0.623	0.020	0.324	2.259

Table D.13. Dispersion for Blocksize of 4 - Write References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	7304	0.030	3.879	0.575	0.603	1.589	2.348
boyer	4769	0.003	3.988	0.187	0.501	1.996	1.993
compile-rb	1075	0.147	3.410	1.162	0.412	2.352	1.559
compile-str	1539	0.111	3.557	1.033	0.381	2.475	1.530
fft	8546	0.002	3.993	0.137	0.630	1.481	2.561
glisp-comp	1705	0.087	3.652	0.893	0.541	1.836	1.894
glisp-pay	1417	0.051	3.797	0.744	0.160	3.361	1.112
qsim	1566	0.051	3.795	0.732	0.473	2.109	1.779
reducer	3899	0.012	3.951	0.359	0.095	3.621	0.874
tmycin	1745	0.028	3.888	0.549	0.411	2.355	1.691
Mean		0.052	3.791	0.637	0.421	2.317	1.734
Std Dev		0.049	0.195	0.343	0.175	0.701	0.512
dec0.001	318	0.325	2.701	1.193	0.441	2.236	1.244
dec1.001	501	0.303	2.788	1.222	0.429	2.285	1.256
dia0	437	0.408	2.366	1.323	0.618	1.526	1.204
forl.000	2395	0.462	2.153	1.265	0.596	1.614	1.150
forl.001	1709	0.225	3.099	1.145	0.539	1.843	1.583
ivex.000 (dup)	1770	0.336	2.657	1.268	0.526	1.895	1.355
ivex.003	681	0.256	2.975	1.153	0.523	1.906	1.503
lisp.000 (dup)	744	0.055	3.781	0.705	0.532	1.874	1.952
lisp.001	823	0.062	3.752	0.721	0.534	1.865	1.933
pasc.001	4686	0.067	3.730	0.699	0.500	2.000	1.739
spic.000 (dup)	991	0.281	2.877	1.189	0.487	2.053	1.346
spic.001	1388	0.437	2.251	1.151	0.541	1.836	0.959
umil1	378	0.417	2.333	1.412	0.619	1.524	1.145
umil2	114	0.274	2.904	1.389	0.581	1.675	1.423
Mean		0.279	2.883	1.131	0.533	1.867	1.414
Std Dev		0.137	0.550	0.244	0.058	0.232	0.298

Table D.14. Dispersion for Blocksize of 8 - Write References

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	3736	0.052	7.584	1.547	0.798	1.614	5.999
boyer	2394	0.007	7.944	0.602	0.751	1.993	5.951
compile-rb	607	0.245	6.040	3.007	0.626	2.988	3.950
compile-str	850	0.195	6.440	2.786	0.608	3.138	4.133
fft	4283	0.004	7.968	0.446	0.815	1.480	6.506
glisp-comp	903	0.138	6.895	2.327	0.769	1.852	5.076
glisp-pay	754	0.108	7.137	2.262	0.282	5.741	3.129
qsim	830	0.105	7.160	2.196	0.728	2.177	5.071
reducer	1964	0.019	7.844	0.960	0.161	6.713	2.643
tmycin	903	0.061	7.513	1.699	0.697	2.426	5.171
Mean		0.093	7.252	1.783	0.624	3.012	4.763
Std Dev		0.081	0.650	0.890	0.224	1.792	1.272
dec0.001	222	0.516	3.869	2.483	0.637	2.905	2.305
dec1.001	346	0.495	4.038	2.710	0.637	2.902	2.390
dia0	302	0.572	3.424	2.723	0.780	1.758	2.151
forl.000	1744	0.630	2.956	2.547	0.767	1.864	2.107
forl.001	1017	0.349	5.208	2.743	0.717	2.265	3.348
ivex.000 (dup)	1198	0.509	3.926	2.749	0.702	2.382	2.622
ivex.003	405	0.375	5.002	2.733	0.655	2.758	2.920
lisp.000 (dup)	392	0.103	7.176	1.965	0.757	1.941	5.275
lisp.001	434	0.111	7.115	2.021	0.759	1.926	5.233
pasc.001	2569	0.150	6.804	2.126	0.747	2.021	4.798
spic.000 (dup)	658	0.458	4.333	2.870	0.698	2.416	2.722
spic.001	1007	0.612	3.103	2.474	0.752	1.982	1.708
umil1	265	0.584	3.328	2.985	0.797	1.623	2.093
umil2	75	0.448	4.413	3.275	0.788	1.693	2.835
Mean		0.422	4.621	2.600	0.728	2.174	3.036
Std Dev		0.182	1.460	0.368	0.055	0.438	1.197

Table D.15. Dispersion for Blocksize of 16 - Write Refcrences

TRACE NAME	number of blocks	percent unused (total trace)	average number referenced	Std Dev	percent unused (first time)	average number referenced	Std Dev
biaslisp	1903	0.069	14.888	3.496	0.896	1.669	13.245
boyer	1206	0.014	15.769	1.802	0.876	1.988	13.781
compile-rb	348	0.342	10.534	6.777	0.754	3.940	8.304
compile-str	476	0.281	11.500	6.498	0.749	4.023	8.974
fft	2151	0.008	15.865	1.345	0.907	1.481	14.393
glisp-comp	484	0.196	12.864	5.556	0.884	1.862	11.020
glisp-pay	402	0.163	13.386	5.537	0.423	9.226	7.637
qsim	456	0.185	13.033	5.759	0.863	2.200	10.940
reducer	994	0.031	15.499	2.503	0.238	12.186	6.668
tmycin	474	0.105	14.312	4.505	0.850	2.392	11.959
Mean		0.139	13.765	4.378	0.744	4.097	10.692
Std Dev		0.114	1.827	1.973	0.229	3.659	2.702
dec0.001	171	0.686	5.023	4.225	0.799	3.222	3.275
dec1.001	252	0.654	5.544	5.095	0.799	3.222	3.679
dia0	219	0.705	4.721	4.989	0.880	1.927	3.328
forl.000	1298	0.752	3.972	4.516	0.878	1.953	3.151
forl.001	609	0.456	8.698	5.941	0.849	2.411	6.573
ivex.000 (dup)	861	0.659	5.462	5.136	0.830	2.719	4.168
ivex.003	261	0.515	7.762	5.746	0.794	3.299	5.222
lisp.000 (dup)	214	0.178	13.145	5.260	0.878	1.958	11.215
lisp.001	237	0.186	13.030	5.248	0.876	1.987	11.074
pasc.001	1375	0.206	12.712	4.321	0.872	2.050	10.679
spic.000 (dup)	421	0.577	6.772	5.652	0.815	2.967	4.866
spic.001	608	0.679	5.140	4.920	0.870	2.084	3.505
umil1	192	0.713	4.594	5.615	0.895	1.682	3.291
umil2	53	0.610	6.245	6.687	0.901	1.585	4.778
Mean		0.541	7.344	5.239	0.853	2.362	5.629
Std Dev		0.206	3.295	0.665	0.038	0.605	3.059

Appendix E. *Blocked* P_{LRU}

Table E.1. LISP All Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
biaslisp	36	0.11	50.28	282	0.89	70.01	1491	4.70	90.00	25283	79.65	95.00	26075	82.14	99.00	31383	98.86	100.00
blk4	32	0.10	50.84	100	0.30	70.39	708	2.12	90.00	1728	5.18	95.06	27644	82.83	99.00	33012	98.91	100.00
blk8	40	0.12	51.13	144	0.42	70.25	368	1.06	90.00	1656	4.78	95.02	27688	79.97	99.00	34232	98.87	100.00
blk16	80	0.22	54.58	224	0.62	70.41	448	1.23	90.55	1184	3.25	95.01	3152	8.65	99.00	36016	98.81	100.00
boyer	45	0.22	50.64	69	0.34	70.08	222	1.09	90.05	838	4.13	95.00	3985	19.65	99.00	12550	61.87	100.00
blk4	32	0.15	50.52	80	0.38	71.27	152	0.72	90.19	232	1.09	95.04	1472	6.93	99.01	13488	63.49	100.00
blk8	32	0.15	53.09	72	0.33	70.17	176	0.80	90.38	272	1.24	95.39	1600	7.29	99.01	14184	64.64	100.00
blk16	48	0.21	60.94	80	0.35	72.80	224	0.99	91.05	320	1.41	95.32	2016	8.91	99.03	14704	64.95	100.00
compile-rb	81	0.46	55.05	108	0.62	70.03	468	2.68	90.03	848	4.85	95.00	3353	19.19	99.00	16716	95.66	100.00
blk4	24	0.09	50.14	176	0.67	70.16	508	1.93	90.00	876	3.33	95.01	2712	10.31	99.00	25276	96.11	100.00
blk8	32	0.10	55.33	160	0.49	70.47	400	1.23	90.16	1104	3.38	95.00	2816	8.63	99.00	30776	94.33	100.00
blk16	48	0.12	55.52	160	0.40	70.17	544	1.36	91.03	1104	2.77	95.01	2960	7.42	99.00	37536	94.03	100.00
compile-str	102	0.51	52.65	274	1.37	70.04	1026	5.12	90.01	2205	11.01	95.01	16795	83.86	99.00	19451	97.12	100.00
blk4	24	0.08	51.04	180	0.62	70.18	828	2.83	90.03	1544	5.28	95.01	13816	47.20	99.00	28028	95.75	100.00
blk8	24	0.07	50.53	144	0.40	70.42	736	2.05	90.04	1496	4.16	95.00	7128	19.84	99.00	33304	92.70	100.00
blk16	48	0.11	57.00	128	0.29	70.91	640	1.46	90.28	1648	3.77	95.01	5824	13.32	99.00	42064	96.23	100.00
fft	28	0.07	54.95	65	0.17	73.17	5432	13.95	90.00	29138	74.82	95.04	29200	74.98	99.02	36052	92.57	100.00
blk4	28	0.07	58.33	72	0.18	72.51	132	0.33	91.69	156	0.39	95.25	29268	73.92	99.02	36196	91.42	100.00
blk8	56	0.13	60.88	136	0.33	70.60	208	0.50	90.04	232	0.56	95.28	15232	36.54	99.00	38408	92.13	100.00
blk16	80	0.19	50.41	224	0.53	70.82	336	0.80	91.35	400	0.95	96.51	5200	12.39	99.05	38704	92.19	100.00
glisp-comp	229	1.57	52.30	447	3.07	70.03	841	5.78	90.02	1409	9.69	95.02	4546	31.25	99.00	14122	97.09	100.00
blk4	28	0.14	51.28	224	1.08	70.06	1120	5.42	90.03	1504	7.27	95.01	4016	19.42	99.00	20188	97.62	100.00
blk8	32	0.12	52.65	112	0.44	70.42	1192	4.66	90.04	1880	7.35	95.03	4648	18.18	99.00	25024	97.90	100.00
blk16	48	0.15	56.58	112	0.34	70.50	1072	3.29	90.10	2192	6.74	95.01	5104	15.68	99.00	31968	98.23	100.00
glisp-pay	135	1.64	50.02	482	5.84	70.02	1638	19.86	90.04	1673	20.28	95.23	1695	20.54	99.11	4708	57.07	100.00
blk4	40	0.39	50.09	148	1.42	70.03	1192	11.48	90.02	2928	28.19	95.23	2996	28.84	99.05	6656	64.07	100.00
blk8	32	0.27	52.87	192	1.62	71.94	1400	11.81	90.03	3160	26.65	95.02	4176	35.22	99.00	6816	57.49	100.00
blk16	48	0.35	56.16	160	1.17	70.24	1168	8.56	90.03	2448	17.94	95.02	5792	42.44	99.15	7984	58.50	100.00
qsim	73	0.64	50.21	224	1.96	70.00	536	4.69	90.00	1461	12.79	95.01	4290	37.55	99.00	9237	80.84	100.00
blk4	16	0.11	55.46	96	0.63	70.15	556	3.67	90.11	860	5.68	95.00	2932	19.37	99.02	12824	84.70	100.00
blk8	24	0.14	58.49	56	0.32	71.32	520	2.93	90.08	840	4.74	95.11	3752	21.17	99.05	15360	86.68	100.00
blk16	32	0.15	52.45	64	0.30	74.22	512	2.43	90.24	912	4.34	95.00	4464	21.23	99.01	16560	78.77	100.00
reducer	14	0.08	51.47	20	0.11	70.45	520	2.78	90.10	809	4.32	95.00	2630	14.05	99.00	16319	87.17	100.00
blk4	12	0.06	51.76	24	0.12	76.75	140	0.67	90.05	712	3.42	95.02	2088	10.03	99.00	17136	82.30	100.00
blk8	16	0.07	52.43	32	0.14	85.09	48	0.22	90.47	408	1.84	95.03	2136	9.63	99.01	16488	74.32	100.00
blk16	32	0.13	68.73	48	0.20	88.47	64	0.27	91.45	208	0.86	95.02	2272	9.42	99.01	15424	63.97	100.00
tmycin	194	1.71	55.22	377	3.32	70.05	636	5.61	90.03	1039	9.16	95.03	2492	21.97	99.00	9196	81.09	100.00
blk4	24	0.17	50.82	352	2.44	72.18	948	6.57	90.08	1136	7.88	95.05	2996	20.78	99.01	12256	84.99	100.00
blk8	24	0.15	55.34	120	0.73	70.18	992	6.05	90.41	1392	8.49	95.00	3360	20.50	99.00	14696	89.65	100.00
blk16	32	0.17	59.47	64	0.35	70.70	768	4.15	90.13	1600	8.64	95.04	3168	17.11	99.00	16816	90.84	100.00

Table E.2. MIT All Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
dec0.001	61	1.01	51.13	127	2.11	70.37	609	10.10	90.02	1148	19.04	95.00	3010	49.92	99.01	5913	98.06	100.00
blk4	12	0.12	50.55	40	0.42	71.76	324	3.36	90.21	864	8.96	95.01	3148	32.63	99.00	9284	96.23	100.00
blk8	16	0.13	53.62	40	0.32	74.72	296	2.35	90.28	608	4.84	95.02	3416	27.18	99.01	12064	95.99	100.00
blk16	32	0.19	60.68	48	0.28	71.22	256	1.51	90.30	720	4.24	95.22	3472	20.43	99.00	16256	95.67	100.00
dec1.001	60	0.65	50.54	127	1.37	70.11	713	7.67	90.02	1454	15.64	95.01	4197	45.14	99.00	9040	97.24	100.00
blk4	12	0.08	50.41	40	0.27	71.91	332	2.25	90.24	940	6.36	95.03	3628	24.55	99.00	14420	97.59	100.00
blk8	16	0.08	53.49	40	0.21	74.36	272	1.42	90.02	744	3.89	95.00	3688	19.30	99.01	18648	97.57	100.00
blk16	32	0.13	60.23	48	0.19	70.78	240	0.95	90.54	720	2.84	95.20	3952	15.58	99.00	24880	98.11	100.00
dia0	150	1.21	50.00	317	2.55	70.95	785	6.32	90.02	2156	17.35	95.00	8741	70.35	99.00	12254	98.62	100.00
blk4	20	0.10	51.37	128	0.61	72.62	648	3.07	90.15	1368	6.49	95.00	11488	54.51	99.00	19936	94.59	100.00
blk8	16	0.06	53.52	72	0.27	70.77	800	2.98	90.20	1288	4.80	95.01	9376	34.98	99.00	25712	95.91	100.00
blk16	32	0.09	60.25	64	0.19	70.45	848	2.46	90.01	1312	3.80	95.05	7616	22.06	99.00	31808	92.12	100.00
forl.000	165	1.02	50.06	340	2.10	70.23	1560	9.64	90.02	1995	12.32	95.00	4232	26.14	99.04	15977	98.69	100.00
blk4	24	0.09	50.49	132	0.49	70.06	804	2.97	90.01	2592	9.56	95.02	5544	20.45	99.00	24172	89.18	100.00
blk8	24	0.07	57.38	80	0.23	70.56	888	2.52	90.09	2176	6.17	95.00	6536	18.53	99.00	31776	90.11	100.00
blk16	32	0.07	56.58	80	0.17	71.22	784	1.67	90.11	1456	3.09	95.00	7184	15.26	99.00	42768	90.86	100.00
forl.001	106	0.66	50.11	261	1.63	70.04	790	4.94	90.00	1484	9.29	95.00	4473	27.99	99.00	15248	95.42	100.00
blk4	16	0.07	50.25	92	0.39	70.31	632	2.67	90.02	1172	4.94	95.00	4364	18.40	99.00	20020	84.42	100.00
blk8	24	0.08	58.05	72	0.25	71.21	576	1.97	90.06	1064	3.65	95.02	3944	13.51	99.01	23832	81.66	100.00
blk16	32	0.09	57.15	64	0.18	71.42	528	1.46	90.15	1136	3.13	95.10	3984	10.98	99.00	29632	81.69	100.00
ivex.000	49	0.16	50.11	158	0.50	70.04	1256	3.98	90.00	3523	11.18	95.00	17209	54.60	99.00	30837	97.84	100.00
blk4	12	0.03	53.38	40	0.09	70.28	428	0.99	90.03	1388	3.21	95.00	12864	29.71	99.00	42344	97.80	100.00
blk8	16	0.03	50.70	32	0.06	71.17	392	0.75	90.09	856	1.64	95.02	10040	19.18	99.00	51216	97.83	100.00
blk16	32	0.05	55.17	48	0.07	72.78	368	0.56	90.01	912	1.39	95.07	8944	13.67	99.00	63952	97.75	100.00
ivex.003	84	1.03	50.29	246	3.01	70.11	1402	17.14	90.08	1592	19.47	95.00	2292	28.03	99.08	7973	97.49	100.00
blk4	16	0.13	52.79	60	0.48	72.58	568	4.58	90.01	1872	15.11	95.00	3000	24.21	99.00	12092	97.58	100.00
blk8	24	0.16	59.88	48	0.31	70.05	464	2.99	90.05	1192	7.69	95.06	3448	22.25	99.02	15120	97.57	100.00
blk16	32	0.16	54.31	64	0.32	71.60	336	1.67	90.04	928	4.60	95.10	4160	20.64	99.00	19664	97.54	100.00
lisp.000	139	2.45	50.75	200	3.52	70.07	397	6.99	90.01	652	11.48	95.00	1805	31.79	99.00	5124	90.24	100.00
blk4	32	0.45	50.06	132	1.85	70.22	448	6.29	90.00	620	8.70	95.03	2192	30.75	99.02	6184	86.76	100.00
blk8	24	0.29	50.61	104	1.24	70.61	448	5.34	90.12	720	8.58	95.08	2072	24.69	99.00	7304	87.03	100.00
blk16	32	0.31	50.70	112	1.08	71.35	544	5.23	90.18	832	8.00	95.13	2112	20.31	99.01	9056	87.08	100.00
lisp.001	139	2.03	50.79	204	2.99	70.14	397	5.81	90.06	590	8.63	95.01	1899	27.79	99.00	6149	89.99	100.00
blk4	32	0.38	50.12	128	1.52	70.02	452	5.35	90.02	616	7.29	95.06	1828	21.65	99.00	6472	76.65	100.00
blk8	24	0.24	50.59	104	1.05	70.32	448	4.54	90.08	720	7.30	95.14	1712	17.37	99.01	7640	77.52	100.00
blk16	32	0.27	51.21	112	0.93	71.28	544	4.51	90.25	832	6.90	95.12	1904	15.78	99.01	9504	78.78	100.00
pasc.001	40	0.21	50.64	73	0.38	71.80	111	0.58	90.09	1417	7.36	95.39	7679	39.88	99.00	19163	99.52	100.00
blk4	12	0.06	58.83	28	0.13	70.56	112	0.52	93.35	120	0.55	95.06	2516	11.58	99.00	21616	99.45	100.00
blk8	24	0.10	64.33	32	0.13	70.87	160	0.66	91.12	168	0.69	96.26	3008	12.32	99.02	24304	99.54	100.00
blk16	32	0.12	51.94	48	0.18	72.44	240	0.88	91.25	272	1.00	96.87	4560	16.75	99.01	27104	99.59	100.00

Table E.2. MIT All Blocked P_{LRU} (Cont'd)

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
spic.000	71	0.92	51.16	122	1.58	70.04	2424	31.44	90.03	3027	39.26	95.00	4353	56.46	99.00	7406	96.06	100.00
blk4	12	0.11	51.94	64	0.57	70.14	444	3.93	90.02	3388	29.96	95.00	6184	54.69	99.01	10724	94.84	100.00
blk8	16	0.11	52.12	48	0.34	70.59	272	1.94	90.77	1496	10.65	95.02	6864	48.86	99.00	13232	94.19	100.00
blk16	32	0.18	59.03	48	0.27	70.44	288	1.62	90.47	704	3.97	95.00	8208	46.26	99.00	16704	94.14	100.00
spic.001	85	1.52	51.66	94	1.68	70.06	417	7.47	90.01	2394	42.87	95.00	4243	75.98	99.00	5336	95.56	100.00
blk4	12	0.14	50.90	72	0.82	70.53	180	2.05	90.81	528	6.02	95.03	5852	66.77	99.00	8368	95.48	100.00
blk8	16	0.15	54.49	48	0.46	71.54	216	2.06	90.03	248	2.37	95.02	6264	59.77	99.00	9952	94.96	100.00
blk16	32	0.25	63.70	48	0.37	70.73	256	1.97	90.09	336	2.59	96.01	5664	43.60	99.00	12208	93.97	100.00
umil1	55	0.48	52.93	148	1.28	70.02	386	3.35	90.00	836	7.26	95.00	4930	42.81	99.01	11267	97.84	100.00
blk4	44	0.23	53.79	124	0.66	71.66	512	2.72	90.09	704	3.74	95.01	4512	23.95	99.00	18492	98.15	100.00
blk8	40	0.17	53.17	144	0.62	70.47	408	1.75	90.05	720	3.08	95.35	3416	14.61	99.00	22952	98.15	100.00
blk16	48	0.16	51.33	128	0.44	70.14	384	1.31	90.15	864	2.95	95.02	2944	10.05	99.01	28752	98.14	100.00
umil2	55	2.46	50.75	137	6.13	70.05	311	13.93	91.31	475	21.27	95.00	995	44.56	99.15	2158	96.64	100.00
blk4	44	1.19	51.18	128	3.46	72.79	440	11.88	90.05	536	14.47	95.04	1296	34.99	99.00	3568	96.33	100.00
blk8	40	0.88	50.61	192	4.23	71.20	408	9.00	90.35	704	15.52	95.48	1256	27.69	99.05	4384	96.65	100.00
blk16	64	1.07	53.01	224	3.75	71.65	400	6.70	90.67	752	12.60	95.05	1376	23.06	99.04	5728	95.98	100.00

Table E.3. LISP Inst Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
biaslisp	296	45.61	50.36	352	54.24	81.95	363	55.93	97.05	363	55.93	97.05	577	88.91	99.12	642	98.92	100.00
blk4	4	0.52	63.56	24	3.09	70.07	468	60.31	94.58	476	61.34	99.08	476	61.34	99.08	772	99.48	100.00
blk8	8	0.86	76.07	8	0.86	76.07	576	62.07	90.35	616	66.38	99.40	616	66.38	99.40	920	99.14	100.00
blk16	16	1.33	82.87	16	1.33	82.87	512	42.67	90.01	800	66.67	96.29	816	68.00	99.66	1184	98.67	100.00
boyer	16	29.63	51.24	20	37.04	70.18	30	55.56	93.21	33	61.11	95.30	50	92.59	99.36	53	98.15	100.00
blk4	8	10.00	51.00	24	30.00	76.41	40	50.00	90.82	52	65.00	97.98	56	70.00	99.47	76	95.00	100.00
blk8	8	7.69	60.23	24	23.08	76.02	56	53.85	94.35	72	69.23	99.66	72	69.23	99.66	96	92.31	100.00
blk16	16	12.50	81.14	16	12.50	81.14	32	25.00	90.38	64	50.00	96.78	80	62.50	99.81	112	87.50	100.00
compile-rb	39	0.58	52.11	119	1.77	70.11	270	4.02	90.02	454	6.77	95.05	1347	20.08	99.09	6673	99.48	100.00
blk4	4	0.04	55.99	48	0.50	70.39	260	2.70	90.04	340	3.54	95.08	1008	10.48	99.00	9572	99.54	100.00
blk8	8	0.07	67.64	24	0.20	73.40	144	1.19	90.28	408	3.37	95.12	1080	8.92	99.01	12048	99.47	100.00
blk16	16	0.11	74.53	16	0.11	74.53	144	0.94	92.03	416	2.73	95.12	1088	7.13	99.01	15200	99.58	100.00
compile-str	80	1.13	50.16	203	2.87	70.14	563	7.96	90.20	1223	17.30	95.03	6744	95.39	99.00	6832	96.63	100.00
blk4	4	0.04	55.92	52	0.52	73.07	340	3.37	90.20	696	6.91	95.02	9200	91.31	99.01	9756	96.82	100.00
blk8	8	0.06	67.75	24	0.19	73.05	304	2.40	90.00	648	5.11	95.02	3056	24.09	99.00	12304	96.97	100.00
blk16	16	0.10	75.45	16	0.10	75.45	192	1.20	90.18	608	3.79	95.32	2256	14.06	99.01	15616	97.31	100.00
fft	29	26.36	96.99	29	26.36	96.99	29	26.36	96.99	29	26.36	96.99	40	36.36	99.12	110	100.00	100.00
blk4	4	3.12	72.34	4	3.12	72.34	32	25.00	99.22	32	25.00	99.22	32	25.00	99.22	128	100.00	100.00
blk8	8	5.26	82.64	8	5.26	82.64	40	26.32	99.55	40	26.32	99.55	40	26.32	99.55	152	100.00	100.00
blk16	16	8.33	89.51	16	8.33	89.51	48	25.00	99.88	48	25.00	99.88	48	25.00	99.88	192	100.00	100.00
glisp-comp	173	4.90	52.21	273	7.73	71.43	465	13.17	90.09	656	18.57	95.08	1224	34.66	99.01	3528	99.89	100.00
blk4	4	0.08	57.12	32	0.65	70.38	416	8.39	90.07	572	11.54	95.06	1280	25.83	99.07	4956	100.00	100.00
blk8	8	0.13	69.18	16	0.25	74.22	424	6.71	90.02	656	10.38	95.03	1472	23.29	99.01	6320	100.00	100.00
blk16	16	0.20	77.26	16	0.20	77.26	320	3.91	90.11	784	9.57	95.19	1680	20.51	99.02	8192	100.00	100.00
glisp-pay	96	12.67	51.37	248	32.72	70.18	753	99.34	97.69	753	99.34	97.69	756	99.74	99.97	[illegible]	100.00	100.00
blk4	4	0.34	55.79	60	5.14	74.37	472	40.41	92.08	1084	92.81	95.18	1164	99.66	99.99	[illegible]	100.00	100.00
blk8	8	0.51	67.48	32	2.04	71.25	376	23.98	90.04	800	51.02	95.17	1560	99.49	99.99	[illegible]	100.00	100.00
blk16	16	0.75	75.70	16	0.75	75.70	208	9.77	90.18	784	36.84	95.05	2128	100.00	100.00	2128	100.00	100.00
qsim	7	0.36	50.77	27	1.37	70.47	136	6.91	90.41	592	30.10	95.37	635	32.28	99.02	1851	94.10	100.00
blk4	8	0.27	75.20	8	0.27	75.20	68	2.31	90.01	184	6.25	95.31	952	32.34	99.62	2796	94.97	100.00
blk8	8	0.21	54.45	16	0.41	82.20	48	1.24	90.28	144	3.71	95.03	1216	31.34	99.02	3720	95.88	100.00
blk16	16	0.32	61.98	32	0.64	88.27	48	0.96	91.14	128	2.56	95.08	800	16.03	99.01	4784	95.83	100.00
reducer	10	0.65	66.34	11	0.72	86.56	72	4.70	90.71	157	10.25	95.11	676	44.13	99.00	1375	89.75	100.00
blk4	8	0.39	71.99	8	0.39	71.99	16	0.78	94.69	20	0.97	95.14	444	21.51	99.00	1872	90.70	100.00
blk8	8	0.30	73.01	8	0.30	73.01	16	0.60	95.83	16	0.60	95.83	432	16.12	99.00	2224	82.98	100.00
blk16	16	0.46	96.11	16	0.46	96.11	16	0.46	96.11	16	0.46	96.11	416	11.82	99.02	2960	84.09	100.00
tmycin	88	6.94	50.05	149	11.74	70.15	291	22.93	90.19	509	40.11	95.07	638	50.28	99.01	1769	100.00	100.00
blk4	4	0.22	56.20	20	1.09	70.92	228	12.47	90.43	324	17.72	95.01	908	49.67	99.01	1828	100.00	100.00
blk8	8	0.34	67.23	16	0.68	74.50	264	11.22	90.59	376	15.99	95.02	1192	50.68	99.07	2312	98.30	100.00
blk16	16	0.54	76.66	16	0.54	76.66	176	5.91	90.02	432	14.52	95.06	1360	45.70	99.01	2944	98.92	100.00

Table E.4. MIT Inst Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
dec0.001	35	1.35	50.24	70	2.69	70.06	401	15.42	90.01	590	22.68	95.01	1284	49.37	99.01	2477	95.23	100.00
blk4	4	0.10	56.17	16	0.41	70.12	156	4.00	90.06	444	11.40	95.01	1164	29.88	99.02	3688	94.66	100.00
blk8	8	0.17	68.23	16	0.35	76.98	104	2.25	90.08	248	5.36	95.15	1064	23.01	99.05	4344	93.95	100.00
blk16	16	0.29	80.92	16	0.29	80.92	64	1.15	90.40	176	3.15	95.19	1040	18.62	99.00	5200	93.12	100.00
dec1.001	35	0.71	52.32	75	1.53	70.05	430	8.77	90.00	655	13.36	95.03	2341	47.75	99.00	4846	98.84	100.00
blk4	4	0.05	55.36	20	0.27	71.12	192	2.58	90.02	504	6.78	95.08	1468	19.74	99.00	6240	83.92	100.00
blk8	8	0.09	67.59	16	0.18	76.65	88	0.98	90.18	256	2.84	95.05	1176	13.07	99.02	7552	83.91	100.00
blk16	16	0.14	79.92	16	0.14	79.92	64	0.57	90.47	192	1.71	95.14	1088	9.70	99.02	9840	87.73	100.00
dia0	222	2.57	50.12	233	2.70	70.30	630	7.29	90.01	2308	26.72	95.04	6480	75.03	99.00	8542	98.91	100.00
blk4	4	0.03	50.93	76	0.55	71.85	436	3.16	90.16	928	6.72	95.01	8868	64.26	99.00	13044	94.52	100.00
blk8	8	0.05	66.23	24	0.14	71.04	528	3.15	92.20	808	4.83	95.01	6440	38.48	99.00	16528	98.76	100.00
blk16	16	0.08	73.33	16	0.08	73.33	432	2.14	90.09	720	3.56	95.03	4656	23.04	99.00	19424	96.12	100.00
forl.000	178	2.60	50.23	285	4.16	70.07	1144	16.70	90.02	1484	21.67	95.00	2445	35.70	99.12	5551	81.05	100.00
blk4	8	0.08	50.82	188	1.76	72.68	1112	10.41	90.03	1864	17.45	95.12	3428	32.10	99.09	8780	82.21	100.00
blk8	8	0.06	65.34	32	0.25	70.36	568	4.43	90.61	2024	15.79	95.00	3304	25.78	99.01	10552	82.33	100.00
blk16	16	0.10	75.40	16	0.10	75.40	480	3.08	90.03	864	5.54	95.11	3328	21.33	99.00	12832	82.26	100.00
forl.001	113	1.27	50.61	202	2.28	70.02	553	6.23	90.00	790	8.90	95.00	1368	15.42	99.05	6434	72.50	100.00
blk4	4	0.03	52.34	80	0.61	70.29	376	2.85	90.01	668	5.06	95.03	1896	14.37	99.00	9272	70.28	100.00
blk8	8	0.05	65.77	24	0.16	70.00	384	2.48	90.30	568	3.67	95.05	1696	10.95	99.01	10824	69.85	100.00
blk16	16	0.09	75.60	16	0.09	75.60	304	1.66	90.34	592	3.24	95.19	1696	9.27	99.00	12624	69.03	100.00
ivex.000	26	0.21	50.35	95	0.76	70.07	581	4.67	90.04	1675	13.46	95.00	7408	59.54	99.00	12142	97.58	100.00
blk4	4	0.02	53.14	8	0.04	71.09	212	1.15	90.26	504	2.73	95.01	5104	27.66	99.00	18004	97.57	100.00
blk8	8	0.04	73.37	8	0.04	73.37	112	0.51	90.00	320	1.46	95.07	3832	17.44	99.01	21432	97.52	100.00
blk16	16	0.06	78.97	16	0.06	78.97	80	0.30	90.22	256	0.97	95.08	2800	10.63	99.00	25664	97.45	100.00
ivex.003	150	3.71	50.74	417	10.31	70.20	1069	26.44	90.80	1166	28.84	95.01	1641	40.59	99.02	3959	97.92	100.00
blk4	4	0.07	54.18	64	1.03	70.36	708	11.42	90.13	1576	25.42	95.09	2040	32.90	99.04	6052	97.61	100.00
blk8	8	0.10	67.93	24	0.31	70.54	368	4.80	90.22	1776	23.15	95.01	2320	30.24	99.00	7480	97.50	100.00
blk16	16	0.17	77.82	16	0.17	77.82	400	4.17	90.02	944	9.85	95.07	2752	28.71	99.02	9328	97.33	100.00
lisp.000	87	11.10	51.04	114	14.54	70.02	191	24.36	90.06	285	36.35	95.01	479	61.10	99.23	650	82.91	100.00
blk4	4	0.34	55.61	44	3.69	70.13	180	15.10	90.11	268	22.48	96.38	536	44.97	99.00	972	81.54	100.00
blk8	8	0.57	69.13	16	1.13	71.48	160	11.30	92.38	240	16.95	95.52	496	35.03	99.02	1128	79.66	100.00
blk16	16	0.90	77.91	16	0.90	77.91	112	6.31	90.52	208	11.71	95.51	400	22.52	99.03	1392	78.38	100.00
lisp.001	87	10.42	51.43	120	14.37	70.10	189	22.64	90.87	268	32.10	95.02	442	52.93	99.03	694	83.11	100.00
blk4	4	0.32	55.66	40	3.15	70.03	188	14.83	90.18	260	20.50	95.01	524	41.33	99.03	1044	82.33	100.00
blk8	8	0.52	69.01	16	1.05	71.37	160	10.47	92.42	240	15.71	95.52	472	30.89	99.00	1232	80.63	100.00
blk16	16	0.83	78.03	16	0.83	78.03	112	5.83	90.66	208	10.83	95.55	384	20.00	99.00	1504	78.33	100.00
pasc.001	42	2.74	86.01	42	2.74	86.01	62	4.05	94.66	102	6.66	95.00	227	14.83	99.13	1530	99.94	100.00
blk4	4	0.17	66.10	56	2.35	95.24	56	2.35	95.24	56	2.35	95.24	224	9.40	99.23	2384	100.00	100.00
blk8	8	0.27	80.37	8	0.27	80.37	64	2.17	97.34	64	2.17	97.34	168	5.71	99.00	2944	100.00	100.00
blk16	16	0.43	87.66	16	0.43	87.66	80	2.17	98.47	80	2.17	98.47	112	3.03	99.25	3696	100.00	100.00

Table E.4. MIT Inst Blocked P_{LRU} (Cont'd)

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
spic.000	49	1.72	50.29	198	6.96	70.04	1806	63.48	90.06	2035	71.53	95.21	2130	74.87	99.09	2727	95.85	100.00
blk4	4	0.09	54.77	52	1.14	70.92	800	17.61	90.07	2820	62.06	95.01	3352	73.77	99.07	4360	95.95	100.00
blk8	8	0.14	66.89	16	0.29	71.18	216	3.88	90.18	2176	39.14	95.00	4112	73.96	99.10	5312	95.54	100.00
blk16	16	0.23	76.22	16	0.23	76.22	160	2.30	90.06	912	13.13	95.01	5072	73.04	99.02	6544	94.24	100.00
spic.001	47	6.24	55.26	51	6.77	89.97	54	7.17	90.02	71	9.43	96.76	417	55.38	99.00	661	87.78	100.00
blk4	4	0.33	56.20	72	5.98	80.84	92	7.64	96.06	92	7.64	96.06	212	17.61	99.03	1052	87.37	100.00
blk8	8	0.56	71.13	8	0.56	71.13	96	6.67	90.33	104	7.22	97.88	136	9.44	99.24	1248	86.67	100.00
blk16	16	0.88	81.02	16	0.88	81.02	112	6.14	93.12	144	7.89	99.02	144	7.89	99.02	1536	84.21	100.00
umil1	31	0.46	50.01	49	0.72	70.13	166	2.44	90.31	616	9.07	95.03	3138	46.20	99.00	6736	99.16	100.00
blk4	4	0.04	55.37	52	0.48	72.16	156	1.43	90.19	276	2.53	95.19	3852	35.26	99.00	10828	99.12	100.00
blk8	8	0.06	65.59	40	0.30	70.13	152	1.14	90.25	280	2.09	95.66	2616	19.56	99.00	13240	98.98	100.00
blk16	16	0.10	75.42	16	0.10	75.42	144	0.88	91.40	352	2.16	95.23	1264	7.75	99.02	16144	98.92	100.00
umil2	33	7.29	60.02	37	8.17	70.25	99	21.85	91.16	123	27.15	95.01	221	48.79	99.01	331	73.07	100.00
blk4	4	0.48	55.04	56	6.70	78.36	124	14.83	90.19	184	22.01	96.03	312	37.32	99.07	624	74.64	100.00
blk8	8	0.69	64.33	80	6.85	70.42	152	13.01	90.32	272	23.29	95.53	400	34.25	99.02	872	74.66	100.00
blk16	16	0.96	73.50	16	0.96	73.50	144	8.65	90.80	320	19.23	95.17	544	32.69	99.02	1248	75.00	100.00

Table E.5. LISP Data Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
biaslisp	28	0.09	51.68	72	0.23	70.14	13948	44.86	90.00	24792	79.73	95.01	25587	82.29	99.00	30736	98.84	100.00
blk4	28	0.09	50.82	92	0.28	70.08	304	0.93	90.06	1264	3.87	95.03	27060	82.83	99.01	32304	98.89	100.00
blk8	40	0.12	52.29	144	0.43	70.64	288	0.85	90.13	1208	3.57	95.02	27712	81.81	99.00	33480	98.84	100.00
blk16	64	0.18	51.79	240	0.67	70.64	416	1.17	90.81	864	2.42	95.03	3088	8.66	99.00	35248	98.79	100.00
boyer	28	0.14	51.53	52	0.26	70.40	697	3.45	90.00	1072	5.30	95.00	5139	25.40	99.00	12497	61.78	100.00
blk4	44	0.21	50.66	84	0.40	72.11	184	0.87	90.33	604	2.85	95.02	2176	10.26	99.00	13456	63.45	100.00
blk8	56	0.26	50.30	120	0.55	71.59	224	1.02	90.03	400	1.82	95.03	1896	8.64	99.00	14176	64.62	100.00
blk16	64	0.28	50.21	144	0.64	71.22	288	1.27	90.10	432	1.91	95.12	2192	9.68	99.02	14704	64.95	100.00
compile-rb	47	0.44	61.62	63	0.59	74.74	230	2.13	90.11	410	3.81	95.00	2235	20.75	99.00	10012	92.95	100.00
blk4	28	0.16	50.37	132	0.74	77.24	320	1.78	90.00	584	3.26	95.04	1940	10.82	99.00	16912	94.29	100.00
blk8	32	0.14	50.89	168	0.71	70.42	328	1.39	90.10	848	3.60	95.11	2136	9.07	99.00	21720	92.25	100.00
blk16	48	0.16	53.46	240	0.78	71.13	464	1.50	90.14	1040	3.36	95.02	2576	8.33	99.01	28560	92.39	100.00
compile-str	47	0.36	53.82	92	0.71	70.11	442	3.41	90.02	1108	8.55	95.00	9937	76.66	99.00	12457	96.10	100.00
blk4	24	0.12	50.44	132	0.64	73.36	532	2.59	90.03	980	4.78	95.01	7092	34.58	99.00	19460	94.89	100.00
blk8	32	0.12	53.08	152	0.58	70.33	584	2.21	90.04	1144	4.33	95.03	6240	23.64	99.00	23984	90.85	100.00
blk16	48	0.14	55.20	192	0.56	70.50	640	1.87	90.06	1376	4.02	95.04	5248	15.33	99.00	32688	95.51	100.00
fft	23	0.06	50.01	28	0.07	71.47	16627	42.81	90.00	29041	74.78	95.06	29094	74.92	99.05	35942	92.55	100.00
blk4	24	0.06	55.34	68	0.17	72.55	96	0.24	90.04	116	0.29	95.05	29160	73.89	99.03	36068	91.39	100.00
blk8	48	0.12	57.30	136	0.33	78.52	168	0.40	90.49	192	0.46	95.55	18688	44.98	99.00	38264	92.10	100.00
blk16	80	0.19	51.21	256	0.61	74.94	304	0.73	90.92	352	0.84	96.17	5248	12.55	99.01	38528	92.15	100.00
glisp-comp	110	1.00	50.03	219	1.99	70.02	464	4.21	90.07	814	7.39	95.00	3553	32.26	99.00	10593	96.19	100.00
blk4	32	0.20	52.05	248	1.52	70.36	824	5.04	90.01	1084	6.63	95.02	3524	21.57	99.00	15852	97.01	100.00
blk8	32	0.15	50.48	168	0.81	70.24	960	4.60	90.05	1528	7.32	95.08	3792	18.17	99.00	20336	97.43	100.00
blk16	48	0.17	52.64	144	0.52	70.57	1088	3.90	90.08	2016	7.22	95.05	4800	17.20	99.00	27328	97.94	100.00
glisp-pay	45	0.60	50.22	215	2.87	70.06	910	12.14	90.04	922	12.31	95.25	943	12.59	99.06	3951	52.73	100.00
blk4	64	0.69	53.91	148	1.58	70.08	1360	14.56	90.00	1908	20.42	95.01	1964	21.02	99.17	5612	60.06	100.00
blk8	56	0.53	50.05	160	1.50	70.82	1216	11.39	90.06	2848	26.69	95.04	3008	28.19	99.06	5632	52.77	100.00
blk16	64	0.52	52.73	208	1.69	70.78	1616	13.15	90.11	3392	27.60	95.03	4464	36.33	99.14	6624	53.91	100.00
qsim	88	0.93	50.09	212	2.24	70.01	570	6.02	90.00	873	9.23	95.00	3534	37.36	99.00	7314	77.31	100.00
blk4	36	0.28	50.45	184	1.44	70.37	568	4.45	90.00	1228	9.63	95.00	2796	21.93	99.00	10492	82.28	100.00
blk8	24	0.16	52.03	176	1.15	70.14	624	4.09	90.05	944	6.18	95.01	3016	19.75	99.00	12960	84.86	100.00
blk16	32	0.17	56.16	144	0.76	70.87	688	3.63	90.17	1008	5.32	95.16	4096	21.64	99.03	14528	76.75	100.00
reducer	7	0.04	50.07	148	0.86	70.60	474	2.76	90.01	742	4.32	95.00	1924	11.19	99.00	14868	86.49	100.00
blk4	8	0.04	66.22	12	0.06	72.05	356	1.86	90.03	788	4.12	95.01	2024	10.58	99.01	13836	72.35	100.00
blk8	16	0.08	74.95	16	0.08	74.95	176	0.86	90.07	848	4.14	95.05	2184	10.67	99.00	15048	73.53	100.00
blk16	32	0.14	80.66	32	0.14	80.66	112	0.50	90.21	672	2.99	95.09	2704	12.03	99.01	14144	62.92	100.00
tmycin	185	1.84	60.57	288	2.86	72.79	428	4.25	90.05	579	5.75	95.00	1732	17.19	99.00	7929	78.72	100.00
blk4	36	0.28	51.01	348	2.69	70.65	796	6.16	90.17	932	7.21	95.02	2272	17.58	99.00	10760	83.26	100.00
blk8	16	0.11	53.18	192	1.29	70.32	872	5.87	90.15	1200	8.07	95.04	2680	18.03	99.00	13168	88.59	100.00
blk16	16	0.09	53.31	80	0.47	70.58	816	4.76	90.05	1520	8.87	95.02	2960	17.27	99.01	15440	90.10	100.00

Table E.6. MIT Data Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
dec0.001	25	0.73	50.22	60	1.74	70.07	261	7.57	90.02	654	18.96	95.00	2198	63.73	99.00	3409	98.84	100.00
blk4	12	0.21	60.09	28	0.48	71.45	184	3.15	90.00	500	8.55	95.03	2192	37.48	99.01	5692	97.33	100.00
blk8	16	0.20	61.44	32	0.39	74.55	200	2.46	90.16	496	6.11	95.03	2648	32.61	99.00	7896	97.24	100.00
blk16	16	0.14	50.80	48	0.41	75.78	272	2.31	90.85	640	5.44	95.09	3248	27.62	99.01	11104	94.42	100.00
dec1.001	24	0.54	50.41	60	1.36	71.42	331	7.49	90.00	790	17.87	95.02	2029	45.90	99.04	4219	95.43	100.00
blk4	8	0.11	50.57	24	0.32	70.23	184	2.46	90.04	604	8.07	95.02	2636	35.20	99.00	7220	96.42	100.00
blk8	16	0.15	61.61	32	0.31	74.90	168	1.61	90.10	600	5.76	95.02	2664	25.58	99.00	10064	96.62	100.00
blk16	16	0.11	50.53	48	0.33	75.55	224	1.51	90.31	704	4.76	95.02	3248	21.97	99.00	14480	97.94	100.00
dia0	17	0.45	50.93	71	1.86	70.08	202	5.30	90.08	473	12.41	95.00	2044	53.62	99.00	3734	97.95	100.00
blk4	16	0.21	54.10	52	0.69	73.83	216	2.86	90.17	492	6.50	95.02	3148	41.62	99.00	6824	90.22	100.00
blk8	16	0.15	52.31	56	0.52	70.99	288	2.66	90.21	576	5.31	95.05	3512	32.40	99.00	9480	87.45	100.00
blk16	32	0.20	59.11	64	0.40	70.47	496	3.10	90.61	688	4.30	95.01	4192	26.20	99.00	13984	87.40	100.00
forl.000	32	0.34	50.76	79	0.84	70.27	310	3.31	90.01	543	5.79	95.03	1593	16.99	99.00	9225	98.40	100.00
blk4	16	0.10	52.34	52	0.31	70.89	256	1.54	90.25	660	3.97	95.01	2360	14.21	99.00	15556	93.64	100.00
blk8	16	0.07	52.03	56	0.24	72.14	320	1.40	90.16	688	3.00	95.03	3096	13.51	99.00	21640	94.42	100.00
blk16	32	0.10	57.04	80	0.25	73.00	432	1.33	90.02	736	2.27	95.01	4384	13.49	99.00	30832	94.88	100.00
forl.001	25	0.35	50.55	69	0.97	70.26	280	3.93	90.00	770	10.81	95.00	3373	47.33	99.00	6929	97.23	100.00
blk4	12	0.11	52.84	40	0.37	71.66	272	2.54	90.00	552	5.15	95.05	3708	34.58	99.00	9764	91.05	100.00
blk8	16	0.11	54.00	40	0.28	70.40	256	1.80	90.14	624	4.39	95.07	3680	25.91	99.00	12576	88.56	100.00
blk16	32	0.17	61.36	48	0.25	70.92	256	1.34	90.16	704	3.69	95.11	2848	14.93	99.01	16768	87.92	100.00
ivex.000	17	0.09	50.30	59	0.31	70.22	756	3.95	90.00	2724	14.24	95.00	10539	55.10	99.00	18747	98.01	100.00
blk4	12	0.05	53.76	40	0.16	70.04	252	1.00	90.09	964	3.82	95.09	8868	35.19	99.00	24696	98.00	100.00
blk8	16	0.05	54.16	48	0.15	71.37	296	0.95	90.15	800	2.57	95.01	8312	26.68	99.00	30536	98.00	100.00
blk16	32	0.08	57.93	80	0.20	72.32	400	0.98	90.16	896	2.20	95.01	7728	18.98	99.00	39856	97.88	100.00
ivex.003	14	0.34	51.80	50	1.20	70.12	190	4.56	90.06	403	9.67	95.10	775	18.60	99.00	4046	97.10	100.00
blk4	8	0.13	50.96	24	0.38	71.49	136	2.16	90.43	308	4.88	95.04	952	15.09	99.01	6156	97.59	100.00
blk8	16	0.20	59.88	32	0.40	72.23	144	1.78	90.43	296	3.66	95.05	1152	14.26	99.01	7872	97.43	100.00
blk16	32	0.29	63.82	48	0.44	71.78	160	1.45	90.25	368	3.33	95.13	1536	13.91	99.00	10784	97.68	100.00
lisp.000	52	1.06	50.04	82	1.68	70.03	230	4.70	90.03	624	12.75	95.01	1688	34.48	99.03	4418	90.25	100.00
blk4	32	0.54	51.32	108	1.81	70.64	276	4.64	90.18	404	6.79	95.03	1940	32.59	99.02	5228	87.84	100.00
blk8	40	0.57	53.98	112	1.60	71.27	344	4.91	90.10	520	7.42	95.02	2472	35.27	99.00	6200	88.47	100.00
blk16	64	0.74	53.22	160	1.84	70.96	448	5.16	90.06	736	8.47	95.18	2704	31.12	99.00	7728	88.95	100.00
lisp.001	52	0.87	50.39	83	1.38	70.02	219	3.65	90.02	474	7.90	95.00	1655	27.59	99.02	5394	89.92	100.00
blk4	32	0.45	51.02	108	1.50	70.62	276	3.83	90.16	396	5.50	95.06	1876	26.07	99.01	4660	64.76	100.00
blk8	40	0.48	53.82	112	1.34	71.31	352	4.21	90.15	520	6.21	95.08	1864	22.27	99.00	5568	66.54	100.00
blk16	64	0.63	53.58	160	1.56	70.94	448	4.38	90.07	720	7.04	95.15	2112	20.66	99.00	7088	69.33	100.00
pasc.001	15	0.08	54.37	20	0.11	71.16	209	1.17	90.00	1288	7.23	95.81	9345	52.47	99.10	17717	99.48	100.00
blk4	8	0.04	56.37	16	0.08	71.76	48	0.25	90.19	72	0.37	95.05	3512	17.96	99.00	19420	99.30	100.00
blk8	16	0.07	57.45	32	0.15	72.86	88	0.40	90.42	104	0.48	95.70	3128	14.35	99.00	21664	99.38	100.00
blk16	16	0.07	51.61	48	0.20	75.90	160	0.67	90.56	192	0.80	96.03	4448	18.48	99.03	23920	99.40	100.00

Table E.6. MIT Data Blocked P_{LRU} (Cont'd)

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
spic.000	28	0.57	50.56	48	0.98	70.26	252	5.16	90.00	953	19.50	95.00	2248	46.01	99.00	4652	95.21	100.00
blk4	8	0.12	54.12	32	0.47	71.64	116	1.69	90.11	484	7.06	95.01	2744	40.02	99.00	6424	93.70	100.00
blk8	8	0.09	50.76	32	0.37	70.46	128	1.46	90.80	360	4.12	95.02	2856	32.69	99.00	8112	92.86	100.00
blk16	16	0.14	56.02	48	0.43	73.69	176	1.56	90.55	336	2.98	95.13	3568	31.63	99.00	10496	93.05	100.00
spic.001	31	0.64	50.46	42	0.87	72.27	1543	31.91	90.00	2677	55.37	95.00	3844	79.50	99.00	4672	96.63	100.00
blk4	8	0.11	50.50	36	0.48	71.09	96	1.27	90.50	2188	28.88	95.00	5564	73.44	99.00	7392	97.57	100.00
blk8	16	0.18	60.16	40	0.44	71.26	128	1.41	91.30	640	7.06	95.00	6456	71.23	99.00	8824	97.35	100.00
blk16	16	0.14	53.66	48	0.43	74.00	160	1.43	90.81	208	1.85	95.27	6864	61.20	99.00	10832	96.58	100.00
umil1	22	0.47	51.57	113	2.39	70.10	223	4.71	90.01	468	9.88	95.07	1812	38.27	99.00	4541	95.90	100.00
blk4	28	0.34	50.84	72	0.89	71.08	356	4.38	91.39	448	5.51	95.03	1852	22.77	99.00	7860	96.65	100.00
blk8	40	0.38	52.73	96	0.91	70.09	296	2.81	90.20	464	4.41	95.03	1712	16.27	99.01	10192	96.88	100.00
blk16	64	0.46	51.13	160	1.14	71.28	304	2.16	90.26	592	4.21	95.07	2016	14.32	99.02	13712	97.39	100.00
umil2	23	1.29	52.45	118	6.63	70.19	214	12.02	90.89	464	26.07	95.14	778	43.71	99.12	1752	98.43	100.00
blk4	36	1.25	50.81	76	2.63	73.54	352	12.21	90.67	392	13.59	95.18	1016	35.23	99.00	2820	97.78	100.00
blk8	40	1.17	51.79	112	3.28	70.62	280	8.20	90.05	456	13.35	95.58	1000	29.27	99.02	3360	98.36	100.00
blk16	80	1.81	54.64	176	3.99	71.92	288	6.52	90.10	512	11.59	95.03	1072	24.27	99.01	4336	98.19	100.00

Table E.7. LISP Read Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
biaslisp	37	0.12	50.87	243	0.80	70.02	23720	77.84	90.01	24170	79.32	95.01	25024	82.12	99.00	30192	99.08	100.00
blk4	72	0.22	51.48	104	0.32	70.84	888	2.72	90.09	1332	4.08	95.00	27156	83.21	99.01	32296	98.96	100.00
blk8	64	0.19	50.23	168	0.50	70.77	480	1.42	90.04	1664	4.92	95.01	28144	83.19	99.00	33472	98.94	100.00
blk16	112	0.31	51.34	304	0.85	72.12	528	1.48	90.02	1632	4.58	95.00	16864	47.33	99.00	35232	98.88	100.00
boyer	27	0.17	50.27	47	0.30	70.62	540	3.46	90.04	745	4.77	95.00	2708	17.35	99.00	9263	59.36	100.00
blk4	44	0.23	51.50	84	0.45	70.82	180	0.96	90.17	784	4.18	95.00	1976	10.52	99.00	11952	63.66	100.00
blk8	64	0.30	54.08	120	0.57	70.79	240	1.14	90.38	584	2.77	95.02	1936	9.17	99.02	13760	65.18	100.00
blk16	80	0.35	58.52	128	0.57	70.14	288	1.27	90.50	448	1.98	95.03	2240	9.90	99.02	14688	64.92	100.00
compile-rb	47	0.45	56.65	63	0.61	72.30	237	2.28	90.04	432	4.16	95.02	2335	22.47	99.00	9635	92.73	100.00
blk4	56	0.31	50.82	132	0.74	74.06	388	2.17	90.04	608	3.40	95.05	2180	12.21	99.00	16840	94.29	100.00
blk8	56	0.24	50.25	216	0.92	70.48	416	1.77	90.14	872	3.71	95.20	2312	9.84	99.00	21680	92.24	100.00
blk16	48	0.16	50.48	288	0.93	70.45	496	1.61	90.10	1152	3.73	95.03	2784	9.02	99.00	28528	92.38	100.00
compile-str	50	0.40	50.61	112	0.91	70.05	478	3.86	90.00	1164	9.41	95.00	9691	78.30	99.00	11987	96.85	100.00
blk4	36	0.18	50.17	132	0.65	70.23	568	2.79	90.04	1116	5.47	95.01	11692	57.34	99.00	19380	95.04	100.00
blk8	32	0.12	50.18	200	0.76	70.58	688	2.61	90.00	1256	4.77	95.03	7112	27.00	99.01	23920	90.80	100.00
blk16	48	0.14	52.82	256	0.75	71.61	736	2.15	90.07	1472	4.31	95.01	6112	17.88	99.00	32656	95.55	100.00
fft	21	0.06	55.29	31	0.08	72.52	24604	66.89	90.00	27002	73.41	95.00	27038	73.50	99.05	34406	93.53	100.00
blk4	28	0.07	50.41	64	0.16	74.74	96	0.24	91.32	3424	8.68	95.01	29172	73.93	99.05	36576	92.69	100.00
blk8	56	0.14	52.71	120	0.29	70.68	168	0.40	90.88	224	0.54	95.05	29440	70.88	99.00	38768	93.34	100.00
blk16	112	0.27	55.68	240	0.57	77.83	304	0.73	90.98	336	0.80	95.18	13728	32.85	99.04	39024	93.38	100.00
glisp-comp	133	1.30	50.02	231	2.26	72.54	465	4.54	90.02	860	8.40	95.03	3434	33.55	99.03	9923	96.96	100.00
blk4	36	0.22	51.04	312	1.92	70.11	872	5.36	90.04	1120	6.88	95.05	3824	23.51	99.00	15784	97.02	100.00
blk8	40	0.19	51.01	200	0.96	70.06	1056	5.07	90.01	1544	7.41	95.04	3936	18.89	99.00	20280	97.35	100.00
blk16	48	0.17	51.73	160	0.57	70.73	1184	4.25	90.06	2112	7.58	95.08	5008	17.97	99.00	27296	97.93	100.00
glisp-pay	59	1.49	50.01	261	6.60	70.54	853	21.57	90.25	862	21.79	95.09	881	22.28	99.04	2458	62.15	100.00
blk4	68	1.08	53.00	196	3.12	70.25	1412	22.51	90.03	1876	29.91	95.57	1908	30.42	99.05	4312	68.75	100.00
blk8	80	0.99	50.26	192	2.38	70.23	1368	16.98	90.02	2888	35.85	95.05	2968	36.84	99.22	4680	58.09	100.00
blk16	80	0.75	52.26	240	2.26	70.69	1680	15.84	90.19	3776	35.60	95.01	4432	41.78	99.07	6336	59.73	100.00
qsim	92	1.04	50.29	216	2.43	70.23	612	6.90	90.01	869	9.80	95.02	3497	39.43	99.00	7020	79.15	100.00
blk4	44	0.35	50.13	196	1.55	70.19	576	4.56	90.14	1468	11.62	95.02	2848	22.54	99.00	10428	82.53	100.00
blk8	24	0.16	51.25	192	1.26	70.09	640	4.21	90.04	1040	6.85	95.02	3064	20.17	99.00	12912	84.99	100.00
blk16	32	0.17	55.86	144	0.76	70.15	704	3.73	90.07	1024	5.42	95.08	4112	21.78	99.03	14496	76.78	100.00
reducer	9	0.05	52.48	205	1.21	71.80	501	2.96	90.00	831	4.91	95.00	2339	13.82	99.02	14631	86.46	100.00
blk4	8	0.04	62.97	16	0.08	74.15	496	2.61	90.02	868	4.57	95.01	2192	11.54	99.00	13776	72.55	100.00
blk8	16	0.08	72.40	16	0.08	72.40	320	1.57	90.34	928	4.55	95.16	2416	11.85	99.00	15024	73.68	100.00
blk16	32	0.14	78.71	32	0.14	78.71	144	0.64	90.13	912	4.06	95.05	2880	12.83	99.02	14112	62.87	100.00
tmycin	178	2.20	57.77	269	3.32	70.02	428	5.29	90.05	605	7.48	95.02	1579	19.52	99.00	6607	81.67	100.00
blk4	48	0.40	50.17	380	3.18	76.33	804	6.72	90.06	944	7.90	95.08	2296	19.20	99.00	10112	84.58	100.00
blk8	16	0.11	52.95	264	1.79	70.09	896	6.06	90.03	1216	8.22	95.03	2776	18.77	99.00	13088	88.48	100.00
blk16	16	0.09	53.11	80	0.47	70.95	912	5.33	90.02	1568	9.17	95.02	3280	19.18	99.00	15408	90.08	100.00

Table E.8. MIT Read Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
dec0.001	37	1.19	50.36	74	2.39	70.50	461	14.86	90.01	839	27.05	95.02	2246	72.41	99.00	3037	97.91	100.00
blk4	12	0.22	50.06	60	1.08	70.21	344	6.20	90.02	972	17.52	95.02	2968	53.50	99.00	5428	97.84	100.00
blk8	24	0.31	58.10	56	0.72	70.40	384	4.91	90.18	960	12.28	95.01	3168	40.53	99.00	7632	97.65	100.00
blk16	48	0.42	64.16	80	0.71	72.30	528	4.65	90.06	1056	9.31	95.04	4032	35.54	99.01	10752	94.78	100.00
dec1.001	31	0.79	50.20	89	2.27	70.03	647	16.46	90.02	917	23.33	95.05	2353	59.87	99.01	3868	98.42	100.00
blk4	12	0.17	50.33	56	0.79	71.19	448	6.29	90.00	1244	17.45	95.03	3368	47.25	99.00	6940	97.36	100.00
blk8	24	0.24	58.51	48	0.48	70.24	440	4.37	90.00	1208	11.99	95.02	3880	38.52	99.01	9792	97.22	100.00
blk16	32	0.22	50.37	64	0.44	70.27	464	3.22	90.16	1280	8.88	95.06	4304	29.86	99.01	14032	97.34	100.00
dia0	34	0.96	50.07	71	2.00	71.32	281	7.93	90.01	573	16.18	95.00	2134	60.25	99.04	3474	98.08	100.00
blk4	44	0.60	51.08	96	1.31	70.17	316	4.32	90.01	772	10.56	95.03	3728	51.01	99.00	6640	90.86	100.00
blk8	40	0.38	54.20	88	0.83	70.60	352	3.31	90.16	864	8.13	95.03	4744	44.62	99.00	9560	89.92	100.00
blk16	48	0.30	51.07	160	1.01	74.74	528	3.34	90.27	1088	6.89	95.06	6096	38.60	99.00	14096	89.26	100.00
forl.000	49	0.67	50.02	92	1.25	70.22	370	5.02	90.02	549	7.45	95.01	1564	21.23	99.00	6649	90.27	100.00
blk4	24	0.18	51.12	96	0.70	70.20	336	2.46	90.05	800	5.86	95.01	2240	16.40	99.00	12684	92.88	100.00
blk8	32	0.17	51.01	96	0.51	70.21	384	2.05	90.00	1008	5.39	95.08	3184	17.02	99.00	17504	93.54	100.00
blk16	48	0.18	51.50	112	0.43	70.67	496	1.89	90.21	1168	4.45	95.02	4576	17.42	99.01	24688	93.97	100.00
forl.001	31	0.55	50.20	89	1.58	70.01	340	6.03	90.02	929	16.47	95.00	2769	49.10	99.00	5435	96.38	100.00
blk4	20	0.21	51.78	60	0.62	70.21	408	4.24	90.10	780	8.10	95.02	3520	36.55	99.00	8764	90.99	100.00
blk8	24	0.18	53.49	72	0.55	70.71	448	3.40	90.02	832	6.31	95.08	4480	33.96	99.00	11128	84.35	100.00
blk16	32	0.18	55.87	80	0.45	70.49	464	2.62	90.06	944	5.34	95.08	4384	24.80	99.00	14752	83.44	100.00
ivex.000	24	0.13	50.22	70	0.39	70.01	1179	6.59	90.00	3134	17.52	95.00	10042	56.12	99.00	17524	97.94	100.00
blk4	12	0.05	50.14	52	0.22	70.16	336	1.40	90.00	1240	5.15	95.01	10572	43.91	99.00	23612	98.07	100.00
blk8	16	0.05	51.79	72	0.24	71.36	328	1.10	90.08	1064	3.57	95.01	9424	31.59	99.00	29224	97.96	100.00
blk16	32	0.08	55.11	96	0.25	72.08	448	1.16	90.03	1104	2.85	95.00	8896	22.94	99.00	37984	97.94	100.00
ivex.003	14	0.43	51.82	59	1.81	71.03	264	8.09	90.01	392	12.01	95.15	791	24.24	99.00	3185	97.61	100.00
blk4	12	0.22	53.82	28	0.51	71.43	184	3.34	90.06	444	8.06	95.01	1124	20.39	99.00	5396	97.90	100.00
blk8	16	0.22	56.35	48	0.65	76.18	216	2.93	90.21	472	6.40	95.01	1320	17.90	99.01	7208	97.72	100.00
blk16	32	0.31	59.05	80	0.78	71.88	272	2.64	90.13	512	4.96	95.05	1760	17.05	99.00	10096	97.83	100.00
lisp.000	62	1.68	50.70	87	2.36	70.05	233	6.31	90.01	607	16.44	95.00	1374	37.21	99.00	3159	85.54	100.00
blk4	44	0.84	51.29	120	2.28	70.35	292	5.54	90.16	432	8.20	95.04	1932	36.67	99.00	4612	87.55	100.00
blk8	48	0.71	52.34	128	1.90	70.49	376	5.59	90.30	536	7.98	95.10	2488	37.02	99.06	5984	89.05	100.00
blk16	80	0.94	55.79	192	2.26	71.63	480	5.65	90.15	768	9.04	95.14	2880	33.90	99.00	7600	89.45	100.00
lisp.001	60	1.37	50.17	89	2.04	70.16	219	5.02	90.03	486	11.13	95.00	1567	35.89	99.00	3338	76.45	100.00
blk4	44	0.70	51.25	124	1.97	70.79	288	4.58	90.05	404	6.43	95.06	1952	31.06	99.00	4136	65.82	100.00
blk8	48	0.60	52.01	128	1.60	70.52	376	4.69	90.05	528	6.59	95.04	2104	26.25	99.01	5336	66.57	100.00
blk16	80	0.80	55.51	192	1.93	71.57	480	4.81	90.16	752	7.54	95.12	2384	23.92	99.02	6864	68.86	100.00
pasc.001	16	0.12	55.12	20	0.15	73.27	60	0.45	90.01	857	6.42	95.33	6283	47.10	99.00	13156	98.61	100.00
blk4	8	0.06	61.26	12	0.09	70.26	48	0.34	95.39	48	0.34	95.39	984	7.00	99.02	13864	98.63	100.00
blk8	16	0.11	64.25	24	0.16	71.80	80	0.55	91.28	88	0.60	96.68	976	6.66	99.23	14440	98.47	100.00
blk16	16	0.10	50.53	32	0.20	74.41	144	0.91	91.34	160	1.01	97.40	400	2.51	99.07	15664	98.49	100.00

Table E.8. MIT Read Blocked P_{LRU} (Cont'd)

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
spic.000	31	0.74	50.23	49	1.17	70.13	600	14.38	90.00	949	22.75	95.00	1995	47.82	99.00	3810	91.32	100.00
blk4	12	0.19	53.54	44	0.71	70.66	184	2.96	90.06	1028	16.56	95.01	2804	45.17	99.02	5732	92.33	100.00
blk8	16	0.20	57.44	56	0.71	71.64	160	2.02	90.10	712	8.97	95.00	3024	38.11	99.00	7336	92.44	100.00
blk16	32	0.31	62.57	64	0.61	70.64	192	1.84	90.47	608	5.84	95.02	3776	36.25	99.00	9632	92.47	100.00
spic.001	34	0.77	53.03	39	0.88	72.00	1851	41.63	90.00	2780	62.53	95.00	3542	79.67	99.00	4316	97.08	100.00
blk4	16	0.24	52.48	60	0.91	72.66	256	3.88	90.06	2464	37.33	95.12	5060	76.67	99.03	6432	97.45	100.00
blk8	16	0.22	54.25	56	0.76	71.07	112	1.51	90.77	2296	30.99	95.00	5032	67.93	99.00	7200	97.19	100.00
blk16	16	0.19	51.87	80	0.95	72.64	144	1.70	90.48	256	3.02	95.09	4832	57.09	99.00	8112	95.84	100.00
umil1	21	0.48	53.61	89	2.04	70.10	175	4.02	90.20	235	5.40	95.00	1586	36.44	99.00	4172	95.86	100.00
blk4	28	0.36	50.66	72	0.93	73.35	332	4.29	91.41	388	5.02	95.06	1676	21.68	99.00	7480	96.74	100.00
blk8	40	0.39	52.03	104	1.02	71.03	296	2.92	90.11	440	4.33	95.09	1608	15.84	99.00	9832	96.85	100.00
blk16	64	0.47	51.81	144	1.05	70.02	304	2.22	90.03	576	4.21	95.62	1808	13.20	99.00	13328	97.31	100.00
umil2	21	1.40	52.51	87	5.81	70.05	175	11.68	92.17	193	12.88	95.24	515	34.38	99.02	1416	94.53	100.00
blk4	32	1.24	51.02	72	2.78	73.22	328	12.67	90.86	364	14.07	95.25	764	29.52	99.00	2216	85.63	100.00
blk8	40	1.27	53.43	112	3.56	73.33	272	8.65	90.27	424	13.49	95.10	752	23.92	99.01	2648	84.22	100.00
blk16	64	1.54	50.65	160	3.86	71.59	288	6.95	90.32	512	12.36	95.05	800	19.30	99.01	3616	87.26	100.00

Table E.9. LISP Write Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
biaslisp	10	0.04	50.82	20991	74.09	70.10	21703	76.60	90.02	21954	77.49	95.00	22098	78.00	99.10	27951	98.65	100.00
blk4	8	0.03	56.52	20	0.07	73.70	272	0.93	90.00	21840	74.75	95.01	22824	78.12	99.03	28832	98.69	100.00
blk8	16	0.05	60.16	32	0.11	74.22	56	0.19	90.01	616	2.06	95.02	23216	77.68	99.01	29496	98.69	100.00
blk16	32	0.11	61.95	64	0.21	77.56	96	0.32	91.01	160	0.53	95.37	22928	75.30	99.05	30064	98.74	100.00
boyer	3	0.02	57.03	5	0.03	73.87	8	0.04	90.95	13	0.07	95.91	31	0.16	99.02	11105	58.40	100.00
blk4	8	0.04	64.06	12	0.06	76.73	20	0.11	97.10	20	0.11	97.10	28	0.15	99.19	11152	58.46	100.00
blk8	16	0.08	72.95	16	0.08	72.95	24	0.12	93.58	32	0.17	99.08	32	0.17	99.08	11208	58.52	100.00
blk16	32	0.17	74.80	32	0.17	74.80	48	0.25	96.47	48	0.25	96.47	64	0.33	99.85	11312	58.62	100.00
compile-rb	9	0.25	72.06	9	0.25	72.06	45	1.23	90.10	142	3.87	95.10	1477	40.29	99.00	3358	91.60	100.00
blk4	16	0.37	50.94	28	0.65	83.06	56	1.30	90.09	92	2.14	95.25	492	11.44	99.00	3984	92.65	100.00
blk8	24	0.49	50.77	48	0.99	84.40	80	1.65	90.20	128	2.64	95.32	400	8.24	99.00	4544	93.57	100.00
blk16	48	0.86	52.88	96	1.72	86.95	128	2.30	90.45	208	3.74	95.97	432	7.76	99.04	5248	94.25	100.00
compile-str	9	0.16	57.23	23	0.42	70.62	142	2.59	90.25	467	8.53	95.01	2837	51.83	99.00	3492	63.79	100.00
blk4	16	0.26	52.74	28	0.46	75.98	84	1.36	90.46	164	2.66	95.01	1604	26.06	99.00	4156	67.51	100.00
blk8	24	0.35	54.14	48	0.71	78.34	120	1.77	91.12	184	2.71	95.16	1032	15.18	99.01	4920	72.35	100.00
blk16	48	0.63	57.02	96	1.26	82.03	176	2.31	91.19	256	3.36	95.40	912	11.98	99.01	5648	74.16	100.00
fft	7	0.02	53.90	21355	62.58	70.00	24911	73.00	90.21	24919	73.02	95.45	24937	73.07	99.46	29763	87.22	100.00
blk4	8	0.02	50.73	20	0.06	76.50	2836	8.30	90.37	24868	72.75	95.01	24940	72.96	99.11	29740	87.00	100.00
blk8	16	0.05	51.52	32	0.09	71.00	64	0.19	91.77	3880	11.32	95.50	24960	72.85	99.06	29776	86.90	100.00
blk16	32	0.09	51.90	64	0.19	71.88	112	0.33	93.89	208	0.60	95.04	24944	72.48	99.10	29840	86.70	100.00
glisp-comp	10	0.16	51.56	27	0.43	70.23	142	2.28	90.02	368	5.91	95.00	2921	46.92	99.04	5889	94.59	100.00
blk4	12	0.18	58.06	28	0.41	72.11	76	1.11	90.18	132	1.93	95.06	1492	21.88	99.00	6476	94.96	100.00
blk8	16	0.22	53.25	40	0.55	72.20	112	1.55	90.87	152	2.10	95.27	944	13.07	99.00	6872	95.13	100.00
blk16	32	0.41	56.18	64	0.83	71.14	176	2.27	90.44	240	3.10	95.52	640	8.26	99.01	7392	95.45	100.00
glisp-pay	35	0.65	71.84	35	0.65	71.84	110	2.04	90.49	143	2.66	99.74	143	2.66	99.74	157	2.92	100.00
blk4	4	0.07	52.53	20	0.35	71.30	84	1.48	90.19	168	2.96	95.19	216	3.81	99.87	256	4.52	100.00
blk8	8	0.13	57.87	24	0.40	72.98	112	1.86	91.02	176	2.92	95.03	288	4.78	99.27	376	6.23	100.00
blk16	16	0.25	60.40	32	0.50	71.03	160	2.49	91.07	240	3.73	95.49	432	6.72	99.59	592	9.20	100.00
qaim	5	0.08	50.72	15	0.25	70.57	82	1.38	90.02	95	1.60	95.73	1312	22.08	99.00	2385	40.13	100.00
blk4	8	0.13	52.49	12	0.19	70.16	48	0.77	90.59	132	2.11	95.06	212	3.38	99.11	2544	40.61	100.00
blk8	16	0.24	62.07	24	0.36	78.40	64	0.96	90.20	176	2.65	95.11	344	5.18	99.01	2736	41.21	100.00
blk16	32	0.44	68.43	48	0.66	83.23	96	1.32	90.67	224	3.07	95.14	608	8.33	99.37	3984	54.61	100.00
reducer	71	0.46	60.97	72	0.47	83.38	81	0.53	91.30	85	0.55	95.18	1458	9.46	99.03	13148	85.34	100.00
blk4	4	0.03	78.71	4	0.03	78.71	72	0.46	90.69	76	0.49	95.54	464	2.97	99.00	10348	66.35	100.00
blk8	8	0.05	84.41	8	0.05	84.41	24	0.15	90.16	80	0.51	96.64	344	2.19	99.00	8440	53.72	100.00
blk16	16	0.10	86.99	16	0.10	86.99	32	0.20	90.19	80	0.50	96.35	240	1.51	99.02	8496	53.42	100.00
tmycin	25	0.37	51.10	47	0.69	71.17	100	1.47	90.12	172	2.53	95.02	834	12.29	99.00	4403	64.90	100.00
blk4	8	0.12	50.26	32	0.46	70.65	96	1.38	90.11	136	1.95	95.08	476	6.82	99.00	4584	65.67	100.00
blk8	16	0.22	58.23	40	0.55	70.66	128	1.77	90.95	176	2.44	95.53	472	6.53	99.02	4800	66.45	100.00
blk16	32	0.42	63.76	64	0.84	72.83	192	2.53	90.17	256	3.38	95.24	576	7.59	99.01	5136	67.72	100.00

Table E.10. MIT Write Blocked P_{LRU}

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
dec0.001	29	3.38	50.24	38	4.42	70.55	123	14.32	90.02	154	17.93	95.06	247	28.75	99.01	793	92.32	100.00
blk4	4	0.31	68.03	8	0.63	70.34	56	4.40	90.70	96	7.55	95.05	248	19.50	99.03	1176	92.45	100.00
blk8	8	0.45	78.42	8	0.45	78.42	48	2.70	90.34	88	4.96	95.63	280	15.77	99.04	1648	92.79	100.00
blk16	16	0.59	83.85	16	0.59	83.85	48	1.75	90.49	96	3.51	95.71	352	12.87	99.04	2384	87.13	100.00
dec1.001	24	1.72	50.13	43	3.08	70.32	126	9.02	90.06	157	11.24	95.13	301	21.55	99.00	1231	88.12	100.00
blk4	4	0.20	67.52	8	0.40	70.07	56	2.79	90.44	100	4.99	95.03	256	12.77	99.01	1788	89.22	100.00
blk8	8	0.29	77.46	8	0.29	77.46	48	1.73	90.14	88	3.18	95.52	280	10.12	99.04	2504	90.46	100.00
blk16	16	0.40	82.91	16	0.40	82.91	48	1.19	90.42	96	2.38	95.37	352	8.73	99.04	3776	93.65	100.00
dia0	11	1.06	50.32	34	3.29	70.57	48	4.64	90.13	74	7.16	95.09	353	34.14	99.01	860	83.17	100.00
blk4	8	0.46	65.09	12	0.69	70.11	56	3.20	90.81	68	3.89	95.19	324	18.54	99.00	1676	95.88	100.00
blk8	8	0.33	63.22	16	0.66	77.31	56	2.32	91.31	80	3.31	95.72	352	14.57	99.01	2072	85.76	100.00
blk16	16	0.46	69.44	32	0.91	83.29	48	1.37	90.51	112	3.20	96.15	400	11.42	99.00	3056	87.22	100.00
forl.000	42	0.82	50.92	74	1.43	70.93	231	4.48	90.00	335	6.50	95.02	827	16.04	99.00	4698	91.12	100.00
blk4	4	0.04	52.84	28	0.29	70.41	128	1.34	90.37	324	3.38	95.05	1312	13.70	99.00	9020	94.15	100.00
blk8	8	0.06	62.49	24	0.17	74.31	152	1.09	91.22	296	2.12	95.06	2152	15.42	99.00	13304	95.36	100.00
blk16	16	0.08	67.60	32	0.15	74.79	208	1.00	90.91	320	1.54	95.07	3872	18.64	99.00	19920	95.92	100.00
forl.001	31	0.59	50.27	57	1.08	70.12	188	3.55	90.02	395	7.46	95.00	2048	38.66	99.00	5158	97.38	100.00
blk4	4	0.06	55.67	24	0.35	71.46	132	1.93	90.18	320	4.68	95.00	2400	35.11	99.01	6608	96.66	100.00
blk8	8	0.10	65.00	24	0.30	73.97	112	1.38	90.09	296	3.64	95.02	2808	34.51	99.00	7808	95.97	100.00
blk16	16	0.16	71.36	16	0.16	71.36	128	1.31	90.67	320	3.28	95.23	2816	28.90	99.00	9216	94.58	100.00
ivex.000	16	0.34	51.13	44	0.94	70.21	328	6.97	90.02	1092	23.22	95.02	2709	57.60	99.00	4517	96.05	100.00
blk4	4	0.06	51.63	16	0.23	71.88	124	1.75	90.08	408	5.76	95.02	2924	41.30	99.00	6764	95.54	100.00
blk8	8	0.08	60.59	16	0.17	70.42	120	1.25	90.50	360	3.76	95.05	3136	32.72	99.00	8640	90.15	100.00
blk16	16	0.12	67.25	32	0.23	75.80	176	1.28	90.53	384	2.79	95.02	3376	24.51	99.01	12384	89.89	100.00
ivex.003	29	1.43	50.30	54	2.67	72.82	238	11.75	90.08	294	14.51	95.06	511	25.22	99.01	1578	77.89	100.00
blk4	4	0.15	59.04	16	0.59	72.68	108	3.96	90.00	268	9.84	95.00	564	20.70	99.01	2120	77.83	100.00
blk8	8	0.25	70.34	8	0.25	70.34	112	3.46	91.58	240	7.41	95.08	648	20.00	99.01	2304	71.11	100.00
blk16	16	0.38	75.15	16	0.38	75.15	112	2.68	90.75	208	4.98	95.05	816	19.54	99.06	2800	67.05	100.00
lisp.000	7	0.25	51.39	20	0.71	70.70	226	8.03	90.05	372	13.22	95.04	1049	37.29	99.01	2518	89.51	100.00
blk4	16	0.54	53.45	32	1.07	70.54	100	3.36	90.02	260	8.74	95.03	564	18.95	99.01	1624	54.57	100.00
blk8	32	1.02	55.17	56	1.79	71.09	128	4.08	90.32	216	6.89	95.02	640	20.41	99.02	1760	56.12	100.00
blk16	48	1.40	53.50	80	2.34	74.87	160	4.67	91.70	224	6.54	95.19	752	21.96	99.14	1936	56.54	100.00
lisp.001	7	0.23	50.47	23	0.75	70.35	203	6.57	90.06	364	11.79	95.01	1122	36.33	99.00	2338	75.71	100.00
blk4	16	0.49	53.93	36	1.09	71.80	112	3.40	90.13	272	8.26	95.04	664	20.17	99.01	1988	60.39	100.00
blk8	32	0.92	55.65	56	1.61	70.07	136	3.92	90.69	232	6.68	95.05	672	19.36	99.02	2104	60.60	100.00
blk16	48	1.27	52.16	80	2.11	73.55	160	4.22	91.29	240	6.33	95.20	752	19.83	99.04	2368	62.45	100.00
pasc.001	14	0.08	57.70	1219	6.97	73.42	5137	29.39	90.00	8684	49.68	95.01	10216	58.45	99.36	17383	99.45	100.00
blk4	4	0.02	52.34	16	0.09	73.69	1632	8.71	90.80	1684	8.98	95.04	10604	56.57	99.07	18616	99.32	100.00
blk8	8	0.04	51.38	32	0.16	72.44	1680	8.17	90.00	2512	12.22	95.19	11576	56.33	99.00	20432	99.42	100.00
blk16	16	0.07	63.04	48	0.22	75.80	128	0.58	90.33	4112	18.69	95.24	14240	64.73	99.00	21872	99.42	100.00

Table E.10. MIT Write Blocked P_{LRU} (Cont'd)

Trace Name	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs	stack size word	% of total words	% of total refs
spic.000	23	0.81	51.20	32	1.12	70.62	218	7.65	90.01	514	18.03	95.08	1121	39.32	99.00	2579	90.46	100.00
blk4	4	0.10	59.41	24	0.61	71.78	56	1.41	90.39	196	4.95	95.04	920	23.21	99.01	3648	92.03	100.00
blk8	8	0.15	71.46	8	0.15	71.46	56	1.06	90.35	128	2.43	95.06	1200	22.80	99.02	4808	91.34	100.00
blk16	16	0.24	77.91	16	0.24	77.91	80	1.19	91.60	144	2.14	95.05	1552	23.04	99.01	6160	91.45	100.00
spic.001	23	0.74	50.95	26	0.83	76.19	964	30.85	90.00	1796	57.47	95.01	2274	72.77	99.02	2993	95.78	100.00
blk4	4	0.07	58.65	20	0.36	71.09	48	0.87	90.94	1868	33.65	95.01	4232	76.22	99.01	5404	97.33	100.00
blk8	8	0.10	68.50	16	0.20	70.07	72	0.89	91.63	2136	26.51	95.00	6472	80.34	99.00	7912	98.21	100.00
blk16	16	0.16	73.90	16	0.16	73.90	96	0.99	91.03	144	1.48	95.01	7920	81.41	99.00	9520	97.86	100.00
umil1	33	3.74	50.23	272	30.84	77.19	281	31.86	90.55	299	33.90	95.66	833	94.44	99.12	843	95.58	100.00
blk4	8	0.53	51.83	44	2.91	71.94	300	19.84	90.06	328	21.69	95.80	608	40.21	99.01	1408	93.12	100.00
blk8	16	0.76	62.80	48	2.26	72.46	104	4.91	90.37	352	16.60	96.31	648	30.57	99.01	1920	90.57	100.00
blk16	32	1.04	71.80	32	1.04	71.80	112	3.65	91.10	224	7.29	95.06	736	23.96	99.01	2704	88.02	100.00
umil2	272	82.17	62.57	275	83.08	71.33	281	84.89	93.88	282	85.20	95.71	287	86.71	99.81	298	90.03	100.00
blk4	40	8.77	59.46	56	12.28	71.39	320	70.17	90.98	332	72.81	96.90	352	77.19	99.77	392	85.97	100.00
blk8	16	2.67	50.92	64	10.67	79.09	112	18.67	90.36	352	58.67	95.23	400	66.67	99.37	496	82.67	100.00
blk16	32	3.77	62.70	64	7.55	76.97	112	13.21	90.84	208	24.53	95.00	448	52.83	99.06	688	81.13	100.00

Appendix F. *Block Transition Probabilities*

Table F.1. Transition Probabilities for Blocksize of 4 - All References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.978	0.211	0.771	0.018	0.022
boyer	0.961	0.180	0.808	0.011	0.039
compile-rb	0.736	0.203	0.786	0.011	0.264
compile-str	0.744	0.204	0.783	0.012	0.256
fft	0.958	0.137	0.842	0.022	0.042
glisp-comp	0.757	0.290	0.701	0.009	0.243
glisp-pay	0.860	0.192	0.803	0.005	0.140
qsim	0.811	0.164	0.829	0.007	0.189
reducer	0.919	0.253	0.736	0.011	0.081
tmycin	0.843	0.236	0.757	0.007	0.157
Mean	0.857	0.207	0.782	0.011	0.143
Std Dev	0.094	0.044	0.043	0.005	0.094
dec0.001	0.880	0.224	0.770	0.006	0.120
dec1.001	0.846	0.217	0.774	0.010	0.154
dia0	0.806	0.199	0.788	0.013	0.194
forl.000	0.884	0.180	0.801	0.019	0.116
forl.001	0.865	0.180	0.806	0.014	0.135
ivex.000 (dup)	0.895	0.138	0.830	0.033	0.105
ivex.003	0.838	0.198	0.795	0.007	0.162
lisp.000 (dup)	0.901	0.172	0.822	0.006	0.099
lisp.001	0.909	0.170	0.822	0.007	0.091
pasc.001	0.960	0.087	0.903	0.010	0.040
spic.000 (dup)	0.823	0.181	0.813	0.007	0.177
spic.001	0.871	0.224	0.772	0.005	0.129
umil1	0.836	0.109	0.880	0.011	0.164
umil2	0.861	0.102	0.895	0.002	0.139
Mean	0.870	0.170	0.819	0.011	0.130
Std Dev	0.040	0.045	0.044	0.008	0.040

Table F.2. Transition Probabilities for Blocksize of 8 - All References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.976	0.182	0.808	0.009	0.024
boyer	0.955	0.147	0.848	0.006	0.045
compile-rb	0.774	0.216	0.777	0.007	0.226
compile-str	0.780	0.226	0.766	0.008	0.220
fft	0.956	0.151	0.837	0.011	0.044
glisp-comp	0.765	0.322	0.672	0.005	0.235
glisp-pay	0.870	0.225	0.772	0.003	0.130
qsim	0.831	0.208	0.788	0.004	0.169
reducer	0.919	0.335	0.659	0.006	0.081
tmycin	0.870	0.287	0.709	0.004	0.130
Mean	0.870	0.230	0.764	0.006	0.130
Std Dev	0.080	0.066	0.065	0.003	0.080
dec0.001	0.912	0.286	0.710	0.004	0.088
dec1.001	0.893	0.272	0.721	0.007	0.107
dia0	0.872	0.228	0.763	0.009	0.128
forl.000	0.926	0.210	0.776	0.013	0.074
forl.001	0.905	0.229	0.762	0.009	0.095
ivex.000 (dup)	0.929	0.191	0.788	0.020	0.071
ivex.003	0.882	0.278	0.718	0.004	0.118
lisp.000 (dup)	0.927	0.204	0.793	0.004	0.073
lisp.001	0.933	0.202	0.793	0.004	0.067
pasc.001	0.968	0.078	0.917	0.006	0.032
spic.000 (dup)	0.871	0.238	0.758	0.004	0.129
spic.001	0.906	0.286	0.711	0.003	0.094
umil1	0.892	0.137	0.856	0.007	0.108
umil2	0.880	0.124	0.875	0.001	0.120
Mean	0.907	0.212	0.781	0.007	0.093
Std Dev	0.028	0.063	0.063	0.005	0.028

Table F.3. Transition Probabilities for Blocksize of 16 - All References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.973	0.195	0.800	0.005	0.027
boyer	0.956	0.180	0.817	0.003	0.044
compile-rb	0.802	0.256	0.740	0.004	0.198
compile-str	0.805	0.263	0.732	0.005	0.195
fft	0.954	0.162	0.833	0.006	0.046
glisp-comp	0.775	0.378	0.618	0.004	0.225
glisp-pay	0.880	0.222	0.777	0.002	0.120
qsim	0.840	0.266	0.732	0.002	0.160
reducer	0.904	0.383	0.614	0.003	0.096
tmycin	0.865	0.365	0.633	0.002	0.135
Mean	0.875	0.267	0.730	0.004	0.125
Std Dev	0.071	0.083	0.082	0.001	0.071
dec0.001	0.940	0.370	0.627	0.003	0.060
dec1.001	0.924	0.351	0.645	0.004	0.076
dia0	0.909	0.285	0.710	0.006	0.091
forl.000	0.948	0.257	0.734	0.009	0.052
forl.001	0.929	0.280	0.714	0.006	0.071
ivex.000 (dup)	0.952	0.223	0.764	0.013	0.048
ivex.003	0.915	0.325	0.672	0.003	0.085
lisp.000 (dup)	0.929	0.247	0.751	0.002	0.071
lisp.001	0.935	0.247	0.751	0.003	0.065
pasc.001	0.969	0.101	0.896	0.003	0.031
spic.000 (dup)	0.901	0.284	0.713	0.003	0.099
spic.001	0.932	0.336	0.662	0.002	0.068
umil1	0.921	0.167	0.829	0.005	0.079
umil2	0.885	0.134	0.865	0.001	0.115
Mean	0.928	0.258	0.738	0.004	0.072
Std Dev	0.022	0.080	0.080	0.003	0.022

Table F.4. Transition Probabilities for Blocksize of 4 - Inst References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.845	0.390	0.607	0.003	0.155
boyer	0.750	0.255	0.745	0.000	0.250
compile-rb	0.817	0.295	0.693	0.013	0.183
compile-str	0.810	0.311	0.676	0.013	0.190
fft	0.906	0.448	0.552	0.000	0.094
glisp-comp	0.805	0.326	0.667	0.007	0.195
glisp-pay	0.791	0.303	0.696	0.001	0.209
qsim	0.774	0.319	0.678	0.003	0.226
reducer	0.828	0.248	0.750	0.002	0.172
tmycin	0.829	0.286	0.710	0.004	0.171
Mean	0.816	0.318	0.677	0.005	0.184
Std Dev	0.042	0.060	0.060	0.005	0.042
dec0.001	0.814	0.306	0.689	0.005	0.186
dec1.001	0.785	0.301	0.690	0.009	0.215
dia0	0.772	0.333	0.653	0.014	0.228
forl.000	0.786	0.309	0.678	0.014	0.214
forl.001	0.786	0.319	0.665	0.016	0.214
ivex.000 (dup)	0.806	0.309	0.670	0.021	0.194
ivex.003	0.775	0.328	0.664	0.007	0.225
lisp.000 (dup)	0.802	0.316	0.682	0.002	0.198
lisp.001	0.807	0.318	0.680	0.002	0.193
pasc.001	0.790	0.335	0.663	0.003	0.210
spic.000 (dup)	0.785	0.343	0.651	0.006	0.215
spic.001	0.804	0.300	0.698	0.001	0.196
umil1	0.766	0.252	0.736	0.012	0.234
umil2	0.760	0.241	0.758	0.001	0.240
Mean	0.788	0.308	0.684	0.008	0.212
Std Dev	0.017	0.029	0.030	0.006	0.017

Table F.5. Transition Probabilities for Blocksize of 8 - Inst References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.853	0.585	0.413	0.002	0.147
boyer	0.692	0.435	0.565	0.000	0.308
compile-rb	0.878	0.441	0.550	0.009	0.122
compile-str	0.871	0.450	0.542	0.009	0.129
fft	0.947	0.654	0.346	0.000	0.053
glisp-comp	0.861	0.474	0.521	0.005	0.139
glisp-pay	0.862	0.463	0.536	0.001	0.138
qsim	0.843	0.415	0.583	0.002	0.157
reducer	0.884	0.472	0.526	0.001	0.116
tmycin	0.878	0.469	0.529	0.003	0.122
Mean	0.857	0.486	0.511	0.003	0.143
Std Dev	0.064	0.075	0.073	0.003	0.064
dec0.001	0.865	0.477	0.520	0.003	0.135
dec1.001	0.843	0.474	0.520	0.006	0.157
dia0	0.846	0.480	0.511	0.009	0.154
forl.000	0.853	0.443	0.548	0.009	0.147
forl.001	0.850	0.448	0.543	0.010	0.150
ivex.000 (dup)	0.868	0.623	0.364	0.013	0.132
ivex.003	0.837	0.460	0.535	0.005	0.163
lisp.000 (dup)	0.859	0.440	0.559	0.001	0.141
lisp.001	0.853	0.438	0.561	0.001	0.147
pasc.001	0.837	0.613	0.385	0.002	0.163
spic.000 (dup)	0.836	0.475	0.522	0.004	0.164
spic.001	0.856	0.452	0.548	0.001	0.144
umil1	0.844	0.424	0.568	0.008	0.156
umil2	0.793	0.417	0.582	0.001	0.207
Mean	0.846	0.476	0.519	0.005	0.154
Std Dev	0.018	0.063	0.065	0.004	0.018

Table F.6. Transition Probabilities for Blocksize of 16 - Inst References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.880	0.699	0.300	0.001	0.120
boyer	0.875	0.625	0.375	0.000	0.125
compile-rb	0.917	0.560	0.435	0.006	0.083
compile-str	0.909	0.570	0.424	0.006	0.091
fft	1.000	0.791	0.209	0.000	0.000
glisp-comp	0.912	0.593	0.404	0.003	0.088
glisp-pay	0.932	0.544	0.455	0.001	0.068
qsim	0.910	0.511	0.488	0.001	0.090
reducer	0.923	0.930	0.069	0.001	0.077
tmycin	0.919	0.592	0.406	0.002	0.081
Mean	0.918	0.641	0.356	0.002	0.082
Std Dev	0.034	0.130	0.129	0.002	0.034
dec0.001	0.899	0.690	0.309	0.002	0.101
dec1.001	0.881	0.677	0.319	0.004	0.119
dia0	0.887	0.568	0.427	0.006	0.113
forl.000	0.894	0.581	0.413	0.006	0.106
forl.001	0.899	0.573	0.421	0.006	0.101
ivex.000 (dup)	0.907	0.707	0.284	0.008	0.093
ivex.003	0.883	0.597	0.400	0.003	0.117
lisp.000 (dup)	0.874	0.598	0.401	0.001	0.126
lisp.001	0.882	0.603	0.397	0.001	0.118
pasc.001	0.861	0.758	0.241	0.001	0.139
spic.000 (dup)	0.880	0.603	0.395	0.003	0.120
spic.001	0.860	0.626	0.373	0.001	0.140
umil1	0.894	0.588	0.407	0.005	0.106
umil2	0.864	0.571	0.429	0.001	0.136
Mean	0.883	0.624	0.372	0.003	0.117
Std Dev	0.015	0.060	0.059	0.003	0.015

Table F.7. Transition Probabilities for Blocksize of 4 - Data References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.980	0.250	0.729	0.021	0.020
boyer	0.961	0.296	0.685	0.019	0.039
compile-rb	0.722	0.239	0.750	0.011	0.278
compile-str	0.736	0.255	0.731	0.013	0.264
fft	0.959	0.148	0.827	0.025	0.041
glisp-comp	0.754	0.357	0.632	0.010	0.246
glisp-pay	0.872	0.251	0.741	0.009	0.128
qsim	0.802	0.278	0.711	0.010	0.198
reducer	0.932	0.337	0.643	0.021	0.068
tmycin	0.847	0.272	0.720	0.008	0.153
Mean	0.856	0.268	0.717	0.015	0.144
Std Dev	0.099	0.057	0.056	0.006	0.099
dec0.001	0.780	0.290	0.703	0.007	0.220
dec1.001	0.797	0.289	0.702	0.009	0.203
dia0	0.793	0.290	0.700	0.011	0.207
forl.000	0.914	0.246	0.730	0.025	0.086
forl.001	0.893	0.251	0.736	0.013	0.107
ivex.000 (dup)	0.910	0.324	0.629	0.047	0.090
ivex.003	0.835	0.301	0.693	0.006	0.166
lisp.000 (dup)	0.894	0.103	0.885	0.012	0.106
lisp.001	0.899	0.103	0.883	0.014	0.101
pasc.001	0.978	0.076	0.911	0.013	0.022
spic.000 (dup)	0.842	0.228	0.765	0.007	0.158
spic.001	0.880	0.271	0.722	0.007	0.120
umil1	0.857	0.193	0.798	0.010	0.143
umil2	0.871	0.181	0.816	0.003	0.129
Mean	0.867	0.225	0.762	0.013	0.133
Std Dev	0.055	0.081	0.084	0.011	0.055

Table F.8. Transition Probabilities for Blocksize of 8 - Data References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.976	0.211	0.778	0.011	0.024
boyer	0.954	0.250	0.740	0.010	0.046
compile-rb	0.745	0.236	0.757	0.008	0.255
compile-str	0.752	0.265	0.726	0.009	0.248
fft	0.957	0.171	0.816	0.013	0.043
glisp-comp	0.748	0.368	0.625	0.007	0.252
glisp-pay	0.865	0.237	0.758	0.005	0.135
qsim	0.814	0.328	0.666	0.006	0.186
reducer	0.927	0.399	0.590	0.011	0.073
tmycin	0.863	0.320	0.675	0.005	0.137
Mean	0.860	0.279	0.713	0.008	0.140
Std Dev	0.092	0.073	0.072	0.003	0.092
dec0.001	0.784	0.410	0.586	0.005	0.216
dec1.001	0.822	0.404	0.589	0.007	0.178
dia0	0.824	0.335	0.657	0.008	0.176
forl.000	0.931	0.329	0.654	0.017	0.069
forl.001	0.909	0.354	0.638	0.008	0.091
ivex.000 (dup)	0.927	0.391	0.579	0.029	0.073
ivex.003	0.824	0.426	0.570	0.004	0.176
lisp.000 (dup)	0.892	0.130	0.863	0.007	0.108
lisp.001	0.906	0.130	0.862	0.008	0.094
pasc.001	0.976	0.078	0.914	0.007	0.024
spic.000 (dup)	0.878	0.330	0.665	0.005	0.122
spic.001	0.894	0.340	0.656	0.004	0.106
umil1	0.863	0.196	0.798	0.006	0.137
umil2	0.855	0.176	0.822	0.002	0.145
Mean	0.877	0.288	0.704	0.008	0.123
Std Dev	0.052	0.119	0.121	0.007	0.052

Table F.9. Transition Probabilities for Blocksize of 16 - Data References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.971	0.227	0.768	0.006	0.029
boyer	0.955	0.172	0.823	0.005	0.045
compile-rb	0.781	0.246	0.749	0.005	0.219
compile-str	0.780	0.279	0.715	0.006	0.220
fft	0.956	0.182	0.811	0.006	0.044
glisp-comp	0.763	0.412	0.583	0.004	0.237
glisp-pay	0.868	0.213	0.784	0.003	0.132
qsim	0.821	0.405	0.591	0.004	0.179
reducer	0.910	0.452	0.542	0.006	0.090
tmycin	0.855	0.412	0.585	0.003	0.145
Mean	0.866	0.300	0.695	0.005	0.134
Std Dev	0.079	0.108	0.108	0.001	0.079
dec0.001	0.795	0.477	0.519	0.004	0.205
dec1.001	0.825	0.473	0.522	0.005	0.175
dia0	0.862	0.417	0.577	0.006	0.138
forl.000	0.934	0.387	0.601	0.012	0.066
forl.001	0.919	0.428	0.566	0.006	0.081
ivex.000 (dup)	0.942	0.435	0.545	0.019	0.058
ivex.003	0.812	0.474	0.524	0.002	0.188
lisp.000 (dup)	0.871	0.157	0.839	0.004	0.129
lisp.001	0.897	0.158	0.837	0.005	0.103
pasc.001	0.969	0.122	0.874	0.004	0.031
spic.000 (dup)	0.888	0.389	0.608	0.003	0.112
spic.001	0.909	0.433	0.564	0.003	0.091
umil1	0.876	0.215	0.781	0.004	0.124
umil2	0.812	0.210	0.789	0.001	0.188
Mean	0.879	0.341	0.653	0.006	0.121
Std Dev	0.054	0.135	0.136	0.005	0.054

Table F.10. Transition Probabilities for Blocksize of 4 - Read References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.967	0.333	0.638	0.029	0.033
boyer	0.946	0.322	0.659	0.019	0.054
compile-rb	0.706	0.280	0.707	0.012	0.294
compile-str	0.718	0.288	0.697	0.015	0.282
fft	0.957	0.238	0.727	0.035	0.043
glisp-comp	0.727	0.399	0.590	0.011	0.273
glisp-pay	0.772	0.261	0.733	0.006	0.228
qsim	0.777	0.288	0.702	0.011	0.223
reducer	0.925	0.400	0.576	0.024	0.075
tmycin	0.828	0.295	0.697	0.008	0.172
Mean	0.832	0.311	0.672	0.017	0.168
Std Dev	0.107	0.054	0.055	0.010	0.107
dec0.001	0.759	0.306	0.683	0.011	0.241
dec1.001	0.767	0.301	0.685	0.014	0.233
dia0	0.746	0.258	0.726	0.015	0.254
forl.000	0.906	0.241	0.727	0.032	0.094
forl.001	0.851	0.200	0.782	0.018	0.149
ivex.000 (dup)	0.899	0.373	0.568	0.059	0.101
ivex.003	0.790	0.288	0.704	0.008	0.210
lisp.000 (dup)	0.879	0.185	0.804	0.012	0.121
lisp.001	0.889	0.186	0.800	0.014	0.111
pasc.001	0.966	0.079	0.908	0.013	0.034
spic.000 (dup)	0.844	0.197	0.793	0.010	0.156
spic.001	0.941	0.200	0.789	0.010	0.059
umil1	0.821	0.207	0.783	0.010	0.179
umil2	0.853	0.194	0.803	0.003	0.147
Mean	0.851	0.230	0.754	0.016	0.149
Std Dev	0.068	0.072	0.080	0.014	0.068

Table F.11. Transition Probabilities for Blocksize of 8 - Read References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.970	0.266	0.720	0.015	0.031
boyer	0.946	0.262	0.727	0.011	0.054
compile-rb	0.730	0.264	0.728	0.008	0.270
compile-str	0.736	0.290	0.700	0.010	0.264
fft	0.957	0.246	0.736	0.018	0.043
glisp-comp	0.737	0.405	0.588	0.007	0.263
glisp-pay	0.796	0.236	0.760	0.004	0.204
qsim	0.801	0.338	0.655	0.007	0.199
reducer	0.919	0.461	0.526	0.013	0.081
tmycin	0.854	0.352	0.643	0.005	0.146
Mean	0.845	0.312	0.678	0.010	0.155
Std Dev	0.097	0.075	0.075	0.005	0.097
dec0.001	0.769	0.379	0.613	0.008	0.231
dec1.001	0.797	0.381	0.609	0.010	0.203
dia0	0.785	0.261	0.727	0.012	0.215
forl.000	0.928	0.276	0.702	0.022	0.072
forl.001	0.879	0.265	0.722	0.013	0.121
ivex.000 (dup)	0.912	0.415	0.549	0.036	0.088
ivex.003	0.787	0.368	0.627	0.005	0.213
lisp.000 (dup)	0.886	0.198	0.795	0.008	0.114
lisp.001	0.903	0.198	0.793	0.009	0.097
pasc.001	0.959	0.074	0.919	0.007	0.041
spic.000 (dup)	0.881	0.254	0.739	0.006	0.119
spic.001	0.924	0.234	0.760	0.006	0.076
umil1	0.831	0.203	0.791	0.006	0.169
umil2	0.837	0.186	0.812	0.002	0.163
Mean	0.863	0.264	0.726	0.011	0.137
Std Dev	0.061	0.095	0.099	0.009	0.061

Table F.12. Transition Probabilities for Blocksize of 16 - Read References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.968	0.284	0.708	0.008	0.032
boyer	0.949	0.181	0.814	0.006	0.051
compile-rb	0.769	0.274	0.720	0.006	0.231
compile-str	0.765	0.304	0.690	0.006	0.235
fft	0.954	0.255	0.736	0.009	0.046
glisp-comp	0.753	0.454	0.541	0.005	0.247
glisp-pay	0.825	0.206	0.791	0.003	0.175
qsim	0.815	0.419	0.577	0.004	0.185
reducer	0.902	0.519	0.475	0.007	0.098
tmycin	0.851	0.456	0.542	0.003	0.149
Mean	0.855	0.335	0.659	0.006	0.145
Std Dev	0.083	0.117	0.117	0.002	0.083
dec0.001	0.774	0.432	0.563	0.006	0.226
dec1.001	0.809	0.430	0.562	0.007	0.191
dia0	0.827	0.294	0.696	0.009	0.173
forl.000	0.934	0.318	0.666	0.016	0.066
forl.001	0.909	0.316	0.675	0.009	0.091
ivex.000 (dup)	0.927	0.437	0.539	0.024	0.073
ivex.003	0.788	0.404	0.592	0.004	0.212
lisp.000 (dup)	0.870	0.214	0.781	0.005	0.130
lisp.001	0.899	0.215	0.779	0.006	0.101
pasc.001	0.947	0.133	0.864	0.004	0.053
spic.000 (dup)	0.883	0.292	0.704	0.004	0.117
spic.001	0.913	0.308	0.689	0.003	0.087
umil1	0.853	0.216	0.779	0.004	0.147
umil2	0.795	0.223	0.776	0.001	0.205
Mean	0.866	0.302	0.690	0.007	0.134
Std Dev	0.059	0.096	0.099	0.006	0.059

Table F.13. Transition Probabilities for Blocksize of 4 - Write References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.972	0.122	0.808	0.070	0.028
boyer	0.997	0.248	0.597	0.155	0.003
compile-rb	0.955	0.307	0.666	0.027	0.045
compile-str	0.960	0.266	0.691	0.043	0.040
fft	0.999	0.136	0.783	0.081	0.001
glisp-comp	0.964	0.186	0.750	0.064	0.036
glisp-pay	0.987	0.312	0.623	0.065	0.013
qsim	0.982	0.284	0.598	0.118	0.018
reducer	0.993	0.560	0.316	0.124	0.007
tmycin	0.987	0.160	0.787	0.053	0.013
Mean	0.980	0.258	0.662	0.080	0.020
Std Dev	0.016	0.127	0.145	0.040	0.016
dec0.001	0.862	0.441	0.555	0.004	0.138
dec1.001	0.866	0.436	0.557	0.007	0.134
dia0	0.888	0.384	0.608	0.008	0.112
forl.000	0.700	0.368	0.602	0.030	0.300
forl.001	0.823	0.371	0.610	0.019	0.177
ivex.000 (dup)	0.801	0.352	0.598	0.050	0.199
ivex.003	0.862	0.421	0.572	0.007	0.138
lisp.000 (dup)	0.938	0.231	0.725	0.044	0.062
lisp.001	0.938	0.210	0.741	0.049	0.062
pasc.001	0.995	0.007	0.942	0.051	0.005
spic.000 (dup)	0.853	0.410	0.578	0.012	0.147
spic.001	0.924	0.388	0.599	0.013	0.076
umil1	0.860	0.244	0.738	0.018	0.140
umil2	0.868	0.081	0.912	0.007	0.132
Mean	0.870	0.310	0.667	0.023	0.130
Std Dev	0.070	0.136	0.128	0.018	0.070

Table F.14. Transition Probabilities for Blocksize of 8 - Write References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.969	0.175	0.791	0.034	0.031
boyer	0.995	0.304	0.624	0.072	0.005
compile-rb	0.957	0.415	0.570	0.015	0.043
compile-str	0.965	0.362	0.614	0.023	0.035
fft	0.997	0.166	0.795	0.039	0.003
glisp-comp	0.969	0.214	0.753	0.033	0.031
glisp-pay	0.980	0.396	0.571	0.033	0.020
qsim	0.984	0.364	0.577	0.059	0.016
reducer	0.994	0.727	0.214	0.059	0.006
tmycin	0.982	0.226	0.747	0.026	0.018
Mean	0.979	0.335	0.626	0.039	0.021
Std Dev	0.014	0.166	0.172	0.018	0.014
dec0.001	0.860	0.635	0.362	0.003	0.140
dec1.001	0.893	0.621	0.374	0.005	0.107
dia0	0.891	0.516	0.479	0.006	0.109
forl.000	0.716	0.505	0.473	0.022	0.284
forl.001	0.844	0.527	0.462	0.011	0.156
ivex.000 (dup)	0.813	0.480	0.487	0.034	0.187
ivex.003	0.874	0.591	0.405	0.004	0.126
lisp.000 (dup)	0.921	0.268	0.709	0.022	0.079
lisp.001	0.924	0.252	0.723	0.025	0.076
pasc.001	0.995	0.011	0.962	0.027	0.005
spic.000 (dup)	0.894	0.573	0.418	0.008	0.106
spic.001	0.919	0.481	0.509	0.009	0.081
umil1	0.868	0.340	0.647	0.013	0.132
umil2	0.840	0.165	0.831	0.005	0.160
Mean	0.875	0.426	0.560	0.014	0.125
Std Dev	0.064	0.189	0.184	0.010	0.064

Table F.15. Transition Probabilities for Blocksize of 16 - Write References

TRACE NAME	P New-Old	P SSD	P NSSD	P Old-New	P New-New
biaslisp	0.972	0.200	0.782	0.017	0.028
boyer	0.990	0.332	0.633	0.035	0.010
compile-rb	0.957	0.428	0.563	0.009	0.043
compile-str	0.964	0.384	0.603	0.013	0.036
fft	0.995	0.183	0.798	0.019	0.005
glisp-comp	0.971	0.234	0.748	0.017	0.029
glisp-pay	0.970	0.437	0.546	0.017	0.030
qsim	0.982	0.405	0.563	0.032	0.018
reducer	0.990	0.805	0.166	0.029	0.010
tmycin	0.979	0.274	0.713	0.014	0.021
Mean	0.977	0.368	0.612	0.020	0.023
Std Dev	0.012	0.180	0.183	0.009	0.012
dec0.001	0.871	0.729	0.269	0.002	0.129
dec1.001	0.893	0.718	0.279	0.004	0.107
dia0	0.913	0.594	0.402	0.004	0.087
forl.000	0.720	0.604	0.380	0.016	0.280
forl.001	0.849	0.612	0.381	0.007	0.151
ivex.000 (dup)	0.816	0.563	0.413	0.024	0.184
ivex.003	0.862	0.676	0.322	0.003	0.138
lisp.000 (dup)	0.930	0.245	0.743	0.012	0.070
lisp.001	0.928	0.234	0.752	0.013	0.072
pasc.001	0.993	0.257	0.729	0.014	0.007
spic.000 (dup)	0.900	0.676	0.319	0.005	0.100
spic.001	0.928	0.583	0.412	0.006	0.072
umil1	0.885	0.454	0.536	0.010	0.115
umil2	0.830	0.306	0.691	0.003	0.170
Mean	0.880	0.518	0.473	0.009	0.120
Std Dev	0.065	0.183	0.180	0.006	0.065

Appendix G. *Original and Sythetic String Entropy Comparisons*

Table G.1. LISP 3SDE1 Differences

	3sdel blk4	3sdel blk8	3sdel blk16	3sdel all
LISP ALL				
1st avg	0.0015	0.0020	0.0017	0.0017
1st std	0.0011	0.0015	0.0015	0.0014
2nd avg	0.0010	0.0011	0.0008	0.0010
2nd std	0.0009	0.0009	0.0006	0.0008
3rd avg	0.0283	0.0357	0.0392	0.0344
3rd std	0.0156	0.0182	0.0192	0.0181
LISP INST				
1st avg	0.0050	0.0043	0.0076	0.0057
1st std	0.0048	0.0042	0.0068	0.0055
2nd avg	0.0024	0.0012	0.0020	0.0019
2nd std	0.0077	0.0010	0.0014	0.0045
3rd avg	0.0388	0.0273	0.0279	0.0313
3rd std	0.0226	0.0433	0.0542	0.0420
LISP DATA				
1st avg	0.0027	0.0025	0.0026	0.0026
1st std	0.0027	0.0019	0.0019	0.0022
2nd avg	0.0014	0.0012	0.0014	0.0013
2nd std	0.0010	0.0009	0.0010	0.0010
3rd avg	0.0489	0.0629	0.0775	0.0631
3rd std	0.0302	0.0282	0.0304	0.0316
LISP READ				
1st avg	0.0023	0.0023	0.0017	0.0021
1st std	0.0023	0.0018	0.0016	0.0019
2nd avg	0.0018	0.0017	0.0015	0.0017
2nd std	0.0016	0.0013	0.0012	0.0014
3rd avg	0.0531	0.0690	0.0823	0.0681
3rd std	0.0515	0.0678	0.0839	0.0693
LISP WRITE				
1st avg	0.0035	0.0032	0.0034	0.0034
1st std	0.0043	0.0034	0.0036	0.0037
2nd avg	0.0026	0.0024	0.0029	0.0027
2nd std	0.0031	0.0032	0.0027	0.0030
3rd avg	0.2004	0.2168	0.2251	0.2141
3rd std	0.1415	0.1481	0.1490	0.1449

Table G.2. MIT 3SDE1 Differences

	3sdel blk4	3sdel blk8	3sdel blk16	3sdel all
MIT ALL				
1st avg	0.0016	0.0017	0.0016	0.0016
1st std	0.0012	0.0015	0.0011	0.0013
2nd avg	0.0014	0.0013	0.0014	0.0014
2nd std	0.0010	0.0011	0.0009	0.0010
3rd avg	0.0394	0.0398	0.0481	0.0425
3rd std	0.0190	0.0222	0.0204	0.0208
MIT INST				
1st avg	0.0038	0.0058	0.0048	0.0048
1st std	0.0040	0.0048	0.0034	0.0042
2nd avg	0.0015	0.0015	0.0017	0.0016
2nd std	0.0011	0.0011	0.0012	0.0012
3rd avg	0.0438	0.0106	0.0101	0.0215
3rd std	0.0384	0.0094	0.0107	0.0283
MIT DATA				
1st avg	0.0023	0.0019	0.0020	0.0021
1st std	0.0019	0.0018	0.0018	0.0018
2nd avg	0.0019	0.0018	0.0018	0.0018
2nd std	0.0017	0.0014	0.0015	0.0015
3rd avg	0.0577	0.0577	0.0664	0.0606
3rd std	0.0621	0.0579	0.0601	0.0597
MIT READ				
1st avg	0.0030	0.0028	0.0028	0.0029
1st std	0.0026	0.0023	0.0023	0.0024
2nd avg	0.0025	0.0026	0.0024	0.0025
2nd std	0.0023	0.0021	0.0017	0.0020
3rd avg	0.0569	0.0584	0.0602	0.0585
3rd std	0.0643	0.0641	0.0649	0.0639
MIT WRITE				
1st avg	0.0042	0.0054	0.0057	0.0051
1st std	0.0043	0.0047	0.0056	0.0049
2nd avg	0.0028	0.0032	0.0041	0.0034
2nd std	0.0026	0.0028	0.0031	0.0028
3rd avg	0.1008	0.0726	0.0724	0.0819
3rd std	0.0664	0.0757	0.0698	0.0714

Table G.3. LISP 4SDE1 Differences

	4sdel blk4	4sdel blk8	4sdel blk16	4sdel all
LISP ALL				
1st avg	0.0019	0.0021	0.0020	0.0020
1st std	0.0012	0.0015	0.0015	0.0014
2nd avg	0.0010	0.0009	0.0008	0.0009
2nd std	0.0007	0.0008	0.0006	0.0007
3rd avg	0.0329	0.0372	0.0339	0.0346
3rd std	0.0132	0.0198	0.0196	0.0177
LISP INST				
1st avg	0.0040	0.0053	0.0050	0.0048
1st std	0.0037	0.0039	0.0046	0.0041
2nd avg	0.0011	0.0012	0.0018	0.0014
2nd std	0.0009	0.0010	0.0013	0.0011
3rd avg	0.0997	0.0307	0.0277	0.0527
3rd std	0.0761	0.0475	0.0520	0.0680
LISP DATA				
1st avg	0.0030	0.0028	0.0025	0.0028
1st std	0.0024	0.0026	0.0021	0.0023
2nd avg	0.0014	0.0013	0.0013	0.0013
2nd std	0.0010	0.0009	0.0010	0.0010
3rd avg	0.0498	0.0530	0.0511	0.0513
3rd std	0.0227	0.0283	0.0272	0.0259
LISP READ				
1st avg	0.0026	0.0034	0.0028	0.0029
1st std	0.0023	0.0022	0.0024	0.0023
2nd avg	0.0017	0.0019	0.0017	0.0018
2nd std	0.0014	0.0014	0.0012	0.0013
3rd avg	0.0617	0.0659	0.0690	0.0655
3rd std	0.0547	0.0568	0.0573	0.0557
LISP WRITE				
1st avg	0.0057	0.0048	0.0053	0.0052
1st std	0.0054	0.0044	0.0046	0.0048
2nd avg	0.0029	0.0031	0.0029	0.0030
2nd std	0.0032	0.0034	0.0034	0.0033
3rd avg	0.1955	0.2056	0.2079	0.2030
3rd std	0.1446	0.1427	0.1450	0.1426

Table G.4. MIT 4SDE1 Differences

	4sdel blk4	4sdel blk8	4sdel blk16	4sdel all
MIT ALL				
1st avg	0.0022	0.0022	0.0018	0.0021
1st std	0.0018	0.0014	0.0014	0.0016
2nd avg	0.0014	0.0014	0.0014	0.0014
2nd std	0.0012	0.0009	0.0010	0.0010
3rd avg	0.0542	0.0364	0.0390	0.0432
3rd std	0.0318	0.0198	0.0203	0.0256
MIT INST				
1st avg	0.0034	0.0039	0.0041	0.0038
1st std	0.0028	0.0033	0.0028	0.0030
2nd avg	0.0014	0.0014	0.0021	0.0016
2nd std	0.0012	0.0012	0.0017	0.0014
3rd avg	0.0318	0.0111	0.0109	0.0179
3rd std	0.0151	0.0097	0.0105	0.0155
MIT DATA				
1st avg	0.0025	0.0025	0.0021	0.0024
1st std	0.0020	0.0027	0.0022	0.0023
2nd avg	0.0019	0.0016	0.0016	0.0017
2nd std	0.0015	0.0015	0.0015	0.0015
3rd avg	0.0894	0.0580	0.0535	0.0670
3rd std	0.0659	0.0583	0.0545	0.0614
MIT READ				
1st avg	0.0032	0.0038	0.0032	0.0034
1st std	0.0026	0.0042	0.0028	0.0033
2nd avg	0.0021	0.0024	0.0020	0.0022
2nd std	0.0016	0.0022	0.0015	0.0018
3rd avg	0.0725	0.0634	0.0602	0.0654
3rd std	0.0731	0.0653	0.0666	0.0681
MIT WRITE				
1st avg	0.0052	0.0061	0.0063	0.0058
1st std	0.0053	0.0054	0.0052	0.0053
2nd avg	0.0030	0.0039	0.0043	0.0037
2nd std	0.0022	0.0028	0.0034	0.0029
3rd avg	0.1457	0.0831	0.0779	0.1022
3rd std	0.0762	0.0877	0.0810	0.0868

Table G.5. LISP 5SDE1 Differences

	5sdel blk4	5sdel blk8	5sdel blk16	5sdel all
LISP ALL				
1st avg	0.0036	0.0048	0.0050	0.0044
1st std	0.0028	0.0036	0.0034	0.0033
2nd avg	0.0178	0.0238	0.0270	0.0229
2nd std	0.0241	0.0249	0.0257	0.0249
3rd avg	0.0192	0.0250	0.0266	0.0236
3rd std	0.0099	0.0195	0.0278	0.0204
LISP INST				
1st avg	0.0047	0.0039	0.0068	0.0052
1st std	0.0039	0.0044	0.0065	0.0051
2nd avg	0.0058	0.0079	0.0102	0.0080
2nd std	0.0047	0.0067	0.0081	0.0068
3rd avg	0.0410	0.0366	0.0324	0.0367
3rd std	0.0431	0.0534	0.0516	0.0491
LISP DATA				
1st avg	0.0044	0.0043	0.0053	0.0047
1st std	0.0037	0.0035	0.0041	0.0037
2nd avg	0.0135	0.0167	0.0197	0.0166
2nd std	0.0256	0.0258	0.0271	0.0260
3rd avg	0.0401	0.0403	0.0357	0.0387
3rd std	0.0248	0.0283	0.0289	0.0272
LISP READ				
1st avg	0.0054	0.0087	0.0126	0.0089
1st std	0.0042	0.0078	0.0135	0.0097
2nd avg	0.0146	0.0241	0.0286	0.0224
2nd std	0.0173	0.0269	0.0337	0.0272
3rd avg	0.0465	0.0484	0.0460	0.0470
3rd std	0.0483	0.0554	0.0580	0.0534
LISP WRITE				
1st avg	0.0189	0.0268	0.0291	0.0250
1st std	0.0323	0.0454	0.0506	0.0432
2nd avg	0.0201	0.0238	0.0233	0.0224
2nd std	0.0283	0.0343	0.0355	0.0326
3rd avg	0.2228	0.2214	0.2240	0.2227
3rd std	0.1419	0.1423	0.1436	0.1410

Table G.6. MIT 5SDE1 Differences

	5sdel blk4	5sdel blk8	5sdel blk16	5sdel all
MIT ALL				
1st avg	0.0037	0.0056	0.0089	0.0061
1st std	0.0028	0.0074	0.0089	0.0071
2nd avg	0.0097	0.0165	0.0246	0.0169
2nd std	0.0081	0.0208	0.0231	0.0194
3rd avg	0.0343	0.0258	0.0231	0.0277
3rd std	0.0217	0.0154	0.0176	0.0189
MIT INST				
1st avg	0.0042	0.0046	0.0049	0.0046
1st std	0.0033	0.0032	0.0040	0.0035
2nd avg	0.0063	0.0099	0.0100	0.0088
2nd std	0.0056	0.0077	0.0089	0.0077
3rd avg	0.0289	0.0256	0.0140	0.0228
3rd std	0.0197	0.0247	0.0133	0.0207
MIT DATA				
1st avg	0.0048	0.0102	0.0108	0.0086
1st std	0.0044	0.0174	0.0161	0.0141
2nd avg	0.0096	0.0196	0.0202	0.0165
2nd std	0.0107	0.0258	0.0288	0.0235
3rd avg	0.0588	0.0405	0.0378	0.0457
3rd std	0.0629	0.0573	0.0535	0.0583
MIT READ				
1st avg	0.0090	0.0081	0.0077	0.0083
1st std	0.0084	0.0077	0.0053	0.0072
2nd avg	0.0091	0.0156	0.0183	0.0143
2nd std	0.0058	0.0229	0.0242	0.0197
3rd avg	0.0614	0.0443	0.0363	0.0473
3rd std	0.0739	0.0625	0.0596	0.0659
MIT WRITE				
1st avg	0.0059	0.0076	0.0112	0.0082
1st std	0.0054	0.0096	0.0155	0.0111
2nd avg	0.0311	0.0396	0.0354	0.0354
2nd std	0.0667	0.0926	0.0679	0.0761
3rd avg	0.0940	0.0880	0.0726	0.0849
3rd std	0.0771	0.0936	0.0822	0.0844

Table G.7. LISP 3SS Differences

	3ssd blk4	3ssd blk8	3ssd blk16	3ssd all
LISP ALL				
1st avg	0.0013	0.0010	0.0012	0.0012
1st std	0.0010	0.0009	0.0010	0.0010
2nd avg	0.0011	0.0008	0.0010	0.0010
2nd std	0.0010	0.0007	0.0008	0.0008
3rd avg	0.0097	0.0102	0.0164	0.0121
3rd std	0.0072	0.0094	0.0175	0.0125
LISP INST				
1st avg	0.0011	0.0009	0.0020	0.0014
1st std	0.0011	0.0011	0.0027	0.0018
2nd avg	0.0012	0.0012	0.0014	0.0012
2nd std	0.0010	0.0012	0.0012	0.0011
3rd avg	0.1158	0.0763	0.0915	0.0946
3rd std	0.1680	0.0511	0.0547	0.1062
LISP DATA				
1st avg	0.0013	0.0013	0.0015	0.0014
1st std	0.0013	0.0010	0.0012	0.0011
2nd avg	0.0011	0.0011	0.0015	0.0012
2nd std	0.0010	0.0010	0.0012	0.0011
3rd avg	0.0161	0.0125	0.0211	0.0166
3rd std	0.0175	0.0069	0.0077	0.0121
LISP READ				
1st avg	0.0021	0.0022	0.0019	0.0021
1st std	0.0019	0.0018	0.0017	0.0018
2nd avg	0.0018	0.0020	0.0017	0.0018
2nd std	0.0015	0.0017	0.0016	0.0016
3rd avg	0.0163	0.0166	0.0245	0.0191
3rd std	0.0178	0.0134	0.0130	0.0152
LISP WRITE				
1st avg	0.0033	0.0046	0.0036	0.0038
1st std	0.0025	0.0033	0.0029	0.0029
2nd avg	0.0031	0.0037	0.0031	0.0033
2nd std	0.0024	0.0029	0.0025	0.0026
3rd avg	0.0849	0.0635	0.0691	0.0725
3rd std	0.0896	0.0650	0.0679	0.0747

Table G.8. MIT 3SS Differences

	3ssd blk4	3ssd blk8	3ssd blk16	3ssd all
MIT ALL				
1st avg	0.0018	0.0018	0.0010	0.0015
1st std	0.0013	0.0015	0.0008	0.0013
2nd avg	0.0017	0.0017	0.0013	0.0016
2nd std	0.0011	0.0016	0.0011	0.0013
3rd avg	0.0173	0.0112	0.0106	0.0130
3rd std	0.0086	0.0073	0.0050	0.0077
MIT INST				
1st avg	0.0018	0.0010	0.0017	0.0015
1st std	0.0011	0.0011	0.0013	0.0012
2nd avg	0.0014	0.0010	0.0014	0.0013
2nd std	0.0011	0.0009	0.0011	0.0010
3rd avg	0.0577	0.0602	0.0694	0.0625
3rd std	0.0581	0.0467	0.0389	0.0484
MIT DATA				
1st avg	0.0020	0.0018	0.0014	0.0017
1st std	0.0016	0.0019	0.0018	0.0018
2nd avg	0.0018	0.0015	0.0015	0.0016
2nd std	0.0014	0.0015	0.0012	0.0014
3rd avg	0.0376	0.0196	0.0207	0.0260
3rd std	0.0234	0.0128	0.0128	0.0188
MIT READ				
1st avg	0.0024	0.0020	0.0030	0.0025
1st std	0.0017	0.0014	0.0018	0.0017
2nd avg	0.0020	0.0018	0.0023	0.0020
2nd std	0.0015	0.0015	0.0016	0.0015
3rd avg	0.0187	0.0183	0.0173	0.0181
3rd std	0.0114	0.0176	0.0180	0.0158
MIT WRITE				
1st avg	0.0042	0.0030	0.0036	0.0036
1st std	0.0050	0.0030	0.0029	0.0037
2nd avg	0.0039	0.0028	0.0037	0.0035
2nd std	0.0042	0.0026	0.0025	0.0032
3rd avg	0.0952	0.0363	0.0388	0.0568
3rd std	0.0803	0.0325	0.0237	0.0582

Table G.9. LISP 4SS Differences

	4ssd blk4	4ssd blk8	4ssd blk16	4ssd all
LISP ALL				
1st avg	0.0016	0.0022	0.0025	0.0021
1st std	0.0013	0.0015	0.0017	0.0015
2nd avg	0.0010	0.0011	0.0011	0.0011
2nd std	0.0010	0.0009	0.0007	0.0009
3rd avg	0.0120	0.0116	0.0142	0.0126
3rd std	0.0064	0.0066	0.0071	0.0067
LISP INST				
1st avg	0.0023	0.0019	0.0024	0.0022
1st std	0.0016	0.0014	0.0034	0.0023
2nd avg	0.0015	0.0012	0.0019	0.0015
2nd std	0.0015	0.0016	0.0020	0.0017
3rd avg	0.0647	0.0837	0.0920	0.0801
3rd std	0.0831	0.0460	0.0332	0.0586
LISP DATA				
1st avg	0.0015	0.0028	0.0018	0.0020
1st std	0.0010	0.0021	0.0014	0.0017
2nd avg	0.0010	0.0016	0.0012	0.0012
2nd std	0.0007	0.0012	0.0011	0.0011
3rd avg	0.0187	0.0157	0.0166	0.0170
3rd std	0.0170	0.0131	0.0100	0.0136
LISP READ				
1st avg	0.0029	0.0031	0.0031	0.0030
1st std	0.0022	0.0025	0.0026	0.0024
2nd avg	0.0018	0.0018	0.0020	0.0019
2nd std	0.0012	0.0017	0.0015	0.0014
3rd avg	0.0247	0.0195	0.0182	0.0208
3rd std	0.0287	0.0139	0.0113	0.0195
LISP WRITE				
1st avg	0.0044	0.0050	0.0050	0.0048
1st std	0.0041	0.0053	0.0039	0.0044
2nd avg	0.0040	0.0034	0.0032	0.0035
2nd std	0.0030	0.0036	0.0025	0.0030
3rd avg	0.1024	0.0722	0.0741	0.0829
3rd std	0.1198	0.0604	0.0665	0.0866

Table G.10. MIT 4SS Differences

	4ssd blk4	4ssd blk8	4ssd blk16	4ssd all
MIT ALL				
1st avg	0.0017	0.0023	0.0018	0.0019
1st std	0.0013	0.0019	0.0014	0.0016
2nd avg	0.0013	0.0017	0.0013	0.0014
2nd std	0.0009	0.0012	0.0009	0.0011
3rd avg	0.0168	0.0090	0.0070	0.0109
3rd std	0.0091	0.0039	0.0041	0.0075
MIT INST				
1st avg	0.0020	0.0018	0.0015	0.0018
1st std	0.0015	0.0013	0.0015	0.0015
2nd avg	0.0011	0.0013	0.0016	0.0013
2nd std	0.0009	0.0010	0.0012	0.0011
3rd avg	0.0630	0.0766	0.0776	0.0724
3rd std	0.0520	0.0603	0.0411	0.0517
MIT DATA				
1st avg	0.0023	0.0026	0.0020	0.0023
1st std	0.0024	0.0023	0.0018	0.0022
2nd avg	0.0015	0.0017	0.0015	0.0016
2nd std	0.0013	0.0016	0.0011	0.0014
3rd avg	0.0313	0.0160	0.0156	0.0209
3rd std	0.0178	0.0085	0.0082	0.0143
MIT READ				
1st avg	0.0036	0.0034	0.0035	0.0035
1st std	0.0032	0.0033	0.0032	0.0032
2nd avg	0.0023	0.0016	0.0018	0.0019
2nd std	0.0017	0.0013	0.0015	0.0015
3rd avg	0.0227	0.0140	0.0118	0.0161
3rd std	0.0153	0.0086	0.0081	0.0120
MIT WRITE				
1st avg	0.0059	0.0035	0.0044	0.0046
1st std	0.0053	0.0035	0.0036	0.0043
2nd avg	0.0038	0.0033	0.0039	0.0037
2nd std	0.0032	0.0023	0.0034	0.0030
3rd avg	0.0676	0.0428	0.0318	0.0474
3rd std	0.0473	0.0319	0.0206	0.0379

Table G.11. LISP 5SS Differences

	5ssd blk4	5ssd blk8	5ssd blk16	5ssd all
LISP ALL				
1st avg	0.0020	0.0035	0.0030	0.0029
1st std	0.0015	0.0016	0.0026	0.0021
2nd avg	0.0009	0.0011	0.0010	0.0010
2nd std	0.0009	0.0009	0.0007	0.0008
3rd avg	0.0119	0.0126	0.0134	0.0126
3rd std	0.0067	0.0135	0.0109	0.0107
LISP INST				
1st avg	0.0043	0.0025	0.0022	0.0030
1st std	0.0070	0.0021	0.0023	0.0045
2nd avg	0.0012	0.0014	0.0016	0.0014
2nd std	0.0015	0.0012	0.0014	0.0014
3rd avg	0.0582	0.0885	0.0902	0.0790
3rd std	0.0835	0.0537	0.0323	0.0614
LISP DATA				
1st avg	0.0025	0.0032	0.0028	0.0028
1st std	0.0013	0.0028	0.0023	0.0022
2nd avg	0.0010	0.0016	0.0014	0.0013
2nd std	0.0007	0.0012	0.0011	0.0010
3rd avg	0.0177	0.0168	0.0213	0.0186
3rd std	0.0136	0.0169	0.0274	0.0200
LISP READ				
1st avg	0.0038	0.0047	0.0041	0.0042
1st std	0.0029	0.0035	0.0034	0.0032
2nd avg	0.0018	0.0021	0.0018	0.0019
2nd std	0.0013	0.0017	0.0015	0.0015
3rd avg	0.0203	0.0138	0.0147	0.0163
3rd std	0.0171	0.0080	0.0121	0.0131
LISP WRITE				
1st avg	0.0088	0.0073	0.0074	0.0079
1st std	0.0087	0.0078	0.0067	0.0077
2nd avg	0.0050	0.0029	0.0045	0.0041
2nd std	0.0035	0.0024	0.0039	0.0034
3rd avg	0.0687	0.0640	0.0628	0.0652
3rd std	0.0705	0.0676	0.0722	0.0694

Table G.12. MIT 5SS Differences

	5ssd blk4	5ssd blk8	5ssd blk16	5ssd all
MIT ALL				
1st avg	0.0028	0.0028	0.0023	0.0026
1st std	0.0018	0.0020	0.0021	0.0020
2nd avg	0.0015	0.0014	0.0010	0.0013
2nd std	0.0010	0.0010	0.0008	0.0010
3rd avg	0.0159	0.0096	0.0068	0.0108
3rd std	0.0111	0.0044	0.0040	0.0082
MIT INST				
1st avg	0.0035	0.0024	0.0015	0.0025
1st std	0.0037	0.0019	0.0014	0.0027
2nd avg	0.0015	0.0012	0.0013	0.0013
2nd std	0.0012	0.0010	0.0013	0.0012
3rd avg	0.0537	0.0764	0.0795	0.0699
3rd std	0.0501	0.0575	0.0372	0.0499
MIT DATA				
1st avg	0.0028	0.0041	0.0031	0.0033
1st std	0.0026	0.0034	0.0025	0.0029
2nd avg	0.0018	0.0019	0.0020	0.0019
2nd std	0.0013	0.0016	0.0014	0.0014
3rd avg	0.0295	0.0179	0.0194	0.0222
3rd std	0.0153	0.0102	0.0099	0.0131
MIT READ				
1st avg	0.0048	0.0045	0.0054	0.0049
1st std	0.0044	0.0035	0.0043	0.0041
2nd avg	0.0021	0.0020	0.0022	0.0021
2nd std	0.0017	0.0016	0.0016	0.0016
3rd avg	0.0231	0.0180	0.0207	0.0206
3rd std	0.0169	0.0124	0.0152	0.0150
MIT WRITE				
1st avg	0.0062	0.0046	0.0072	0.0060
1st std	0.0074	0.0050	0.0102	0.0078
2nd avg	0.0032	0.0033	0.0035	0.0033
2nd std	0.0029	0.0026	0.0030	0.0028
3rd avg	0.0663	0.0466	0.0355	0.0495
3rd std	0.0484	0.0385	0.0243	0.0402

Table G.13. LISP 3SDSS Differences

	3sdss blk4	3sdss blk8	3sdss blk16	3sdss all
LISP ALL				
1st avg	0.0016	0.0016	0.0016	0.0016
1st std	0.0016	0.0010	0.0010	0.0012
2nd avg	0.0008	0.0009	0.0009	0.0009
2nd std	0.0006	0.0007	0.0007	0.0007
3rd avg	0.0482	0.0392	0.0411	0.0428
3rd std	0.0336	0.0305	0.0298	0.0312
LISP INST				
1st avg	0.0052	0.0055	0.0060	0.0056
1st std	0.0062	0.0063	0.0060	0.0061
2nd avg	0.0020	0.0019	0.0025	0.0022
2nd std	0.0021	0.0019	0.0023	0.0021
3rd avg	0.0181	0.0114	0.0080	0.0125
3rd std	0.0166	0.0084	0.0057	0.0119
LISP DATA				
1st avg	0.0026	0.0022	0.0024	0.0024
1st std	0.0018	0.0020	0.0020	0.0019
2nd avg	0.0016	0.0014	0.0014	0.0015
2nd std	0.0012	0.0009	0.0012	0.0011
3rd avg	0.0629	0.0669	0.0667	0.0655
3rd std	0.0406	0.0392	0.0353	0.0381
LISP READ				
1st avg	0.0017	0.0020	0.0020	0.0019
1st std	0.0014	0.0016	0.0015	0.0015
2nd avg	0.0012	0.0012	0.0010	0.0012
2nd std	0.0010	0.0012	0.0008	0.0010
3rd avg	0.0494	0.0482	0.0458	0.0478
3rd std	0.0278	0.0226	0.0191	0.0232
LISP WRITE				
1st avg	0.0036	0.0039	0.0040	0.0038
1st std	0.0043	0.0039	0.0045	0.0042
2nd avg	0.0033	0.0028	0.0030	0.0030
2nd std	0.0031	0.0023	0.0027	0.0027
3rd avg	0.1759	0.1761	0.1730	0.1750
3rd std	0.1366	0.1365	0.1346	0.1344

Table G.14. MIT 3SDSS Differences

	3sdss blk4	3sdss blk8	3sdss blk16	3sdss all
MIT ALL				
1st avg	0.0015	0.0014	0.0011	0.0014
1st std	0.0012	0.0011	0.0010	0.0011
2nd avg	0.0013	0.0012	0.0011	0.0012
2nd std	0.0010	0.0009	0.0007	0.0009
3rd avg	0.0339	0.0372	0.0417	0.0376
3rd std	0.0136	0.0182	0.0189	0.0172
MIT INST				
1st avg	0.0059	0.0059	0.0068	0.0062
1st std	0.0044	0.0046	0.0053	0.0048
2nd avg	0.0020	0.0020	0.0016	0.0019
2nd std	0.0015	0.0017	0.0013	0.0015
3rd avg	0.0158	0.0066	0.0048	0.0091
3rd std	0.0129	0.0046	0.0032	0.0094
MIT DATA				
1st avg	0.0020	0.0024	0.0022	0.0022
1st std	0.0017	0.0016	0.0015	0.0016
2nd avg	0.0015	0.0016	0.0020	0.0017
2nd std	0.0011	0.0013	0.0017	0.0014
3rd avg	0.0519	0.0515	0.0546	0.0527
3rd std	0.0550	0.0549	0.0560	0.0549
MIT READ				
1st avg	0.0021	0.0019	0.0022	0.0021
1st std	0.0015	0.0017	0.0022	0.0018
2nd avg	0.0015	0.0016	0.0018	0.0016
2nd std	0.0012	0.0013	0.0012	0.0013
3rd avg	0.0495	0.0483	0.0479	0.0486
3rd std	0.0544	0.0565	0.0570	0.0555
MIT WRITE				
1st avg	0.0058	0.0065	0.0075	0.0066
1st std	0.0048	0.0051	0.0066	0.0056
2nd avg	0.0036	0.0036	0.0041	0.0038
2nd std	0.0024	0.0028	0.0031	0.0028
3rd avg	0.0624	0.0620	0.0603	0.0616
3rd std	0.0617	0.0619	0.0557	0.0594

Table G.15. LISP 4SDSS Differences

	4sdss blk4	4sdss blk8	4sdss blk16	4sdss all
LISP ALL				
1st avg	0.0020	0.0014	0.0019	0.0018
1st std	0.0016	0.0012	0.0015	0.0014
2nd avg	0.0008	0.0008	0.0012	0.0009
2nd std	0.0007	0.0005	0.0010	0.0008
3rd avg	0.0870	0.0684	0.0938	0.0897
3rd std	0.0639	0.0718	0.0705	0.0681
LISP INST				
1st avg	0.0035	0.0055	0.0048	0.0046
1st std	0.0038	0.0067	0.0049	0.0053
2nd avg	0.0014	0.0022	0.0015	0.0017
2nd std	0.0012	0.0027	0.0014	0.0019
3rd avg	0.0905	0.0644	0.0537	0.0695
3rd std	0.0611	0.0860	0.1000	0.0844
LISP DATA				
1st avg	0.0021	0.0020	0.0022	0.0021
1st std	0.0014	0.0017	0.0018	0.0017
2nd avg	0.0010	0.0015	0.0013	0.0013
2nd std	0.0009	0.0010	0.0011	0.0010
3rd avg	0.1148	0.1283	0.1400	0.1277
3rd std	0.0738	0.0818	0.0805	0.0786
LISP READ				
1st avg	0.0017	0.0015	0.0014	0.0015
1st std	0.0015	0.0013	0.0013	0.0014
2nd avg	0.0016	0.0012	0.0013	0.0014
2nd std	0.0011	0.0010	0.0009	0.0010
3rd avg	0.1258	0.1447	0.1529	0.1411
3rd std	0.1026	0.1238	0.1395	0.1221
LISP WRITE				
1st avg	0.0051	0.0050	0.0041	0.0048
1st std	0.0039	0.0043	0.0029	0.0037
2nd avg	0.0045	0.0032	0.0034	0.0037
2nd std	0.0043	0.0037	0.0027	0.0036
3rd avg	0.2984	0.2862	0.2853	0.2900
3rd std	0.1515	0.1389	0.1320	0.1396

Table G.16. MIT 4SDSS Differences

	4sdss blk4	4sdss blk8	4sdss blk16	4sdss all
MIT ALL				
1st avg	0.0022	0.0018	0.0023	0.0021
1st std	0.0016	0.0014	0.0017	0.0016
2nd avg	0.0018	0.0016	0.0015	0.0016
2nd std	0.0014	0.0012	0.0012	0.0013
3rd avg	0.0645	0.0697	0.0815	0.0719
3rd std	0.0162	0.0216	0.0239	0.0219
MIT INST				
1st avg	0.0038	0.0058	0.0051	0.0049
1st std	0.0034	0.0046	0.0044	0.0042
2nd avg	0.0017	0.0017	0.0022	0.0019
2nd std	0.0012	0.0012	0.0016	0.0014
3rd avg	0.0796	0.0306	0.0247	0.0449
3rd std	0.0452	0.0126	0.0185	0.0380
MIT DATA				
1st avg	0.0020	0.0026	0.0020	0.0022
1st std	0.0016	0.0021	0.0014	0.0017
2nd avg	0.0016	0.0019	0.0023	0.0019
2nd std	0.0012	0.0012	0.0018	0.0014
3rd avg	0.0951	0.0869	0.1003	0.0941
3rd std	0.0527	0.0598	0.0703	0.0611
MIT READ				
1st avg	0.0024	0.0022	0.0027	0.0024
1st std	0.0020	0.0019	0.0022	0.0020
2nd avg	0.0017	0.0017	0.0024	0.0020
2nd std	0.0016	0.0014	0.0020	0.0017
3rd avg	0.0877	0.0780	0.0887	0.0848
3rd std	0.0576	0.0575	0.0685	0.0611
MIT WRITE				
1st avg	0.0052	0.0057	0.0061	0.0057
1st std	0.0043	0.0048	0.0052	0.0048
2nd avg	0.0033	0.0044	0.0049	0.0042
2nd std	0.0024	0.0039	0.0038	0.0034
3rd avg	0.1427	0.1050	0.0980	0.1152
3rd std	0.0824	0.0920	0.0770	0.0857

Bibliography

AHH89. A. Agarwal, M. Horowitz, and J. Hennessy. An analytical cache model. *ACM Transactions on Computer Systems*, 7(2):184–215, May 1989.

ASH86. A. Agarwal, R. L. Sites, and M. Horowitz. Atum: a new technique for capturing address traces using microcode. In *Proc. of the 13th Annual Symposium on Computer Architecture*, pages 119–127, New York NY (USA), June 1986. IEEE.

BBK77. A.P. Batson, D.W.E. Blatt, and J.P. Kearns. Structure within locality intervals. In H. Beilner and E.Gelenbe, editors, *Measuring, Modelling, and Evaluating Computer Systems*, pages 221–232. North-Holland Publishing Company, 1977.

Bel66. L.A. Belady. A study of replacement algorithms for a virtual-storage computer. *IBM Systems Journal*, 5(2):78–101, 1966.

BF75. R.P. Bogott and M.A. Franklin. Evaluation of markov program models in virtual memory systems. *Software Practice and Experience*, 5:337–346, 1975.

Ble92. Michael E. Bletzinger. An investigation of structural locality in the memory referencing behavior of software. Master's thesis, Air Force Institute of Technology, Wright-Patterson AFB, OH USA, December 1992.

BM84. Richard B. Bunt and Jennifer M. Murphy. The measurement of locality and the behaviour of programs. *Computer Journal*, 27(3):238–245, 1984.

BMM84. Richard B. Bunt, Jennifer M. Murphy, and Shikharesh Majunmdar. A measure of program locality and its application. In *Proceedings of the 1984 Conference on Measurement and Modeling of Computer Systems*, pages 28–39, 1984.

CO76. Wesley W. Chu and Holger Opderbeck. Program behavior and the page-fault-frequency replacement algorithm. *IEEE Computer*, pages 29–38, November 1976.

Den68. Peter J. Denning. The working-set model for program behavior. *Communications of the ACM*, 11(5):323–333, May 1968.

DS72. Peter J. Denning and Stuart C. Schwartz. On modeling program behavior. In *Spring Joint Computer Conference*, pages 943–937, 1972.

Fer75. Domenico Ferrari. Tailoring programs to models of program behavior. *IBM Journal of Research and Development*, pages 244–251, May 1975.

FR90. C. Fricker and P. Robert. Memory reference model for the cache memories analysis. In *Performance'90: Proceedings of the 14th IFIP WG 7.3 International Symposium on Computer Performance*, pages 255–269, Edinburgh, Scotland, September 1990.

FR91. C. Fricker and P. Robert. An analytical cache model. Technical Report 1496, INRIA, Domaine de Voluceau, Rocquencourt B.P. 105 78153 Le Chesnay Cédex France, September 1991.

Har88. Stephen J. Hartley. An analysis of some problems in managing virtual memory systems with fast secondary storage devices. *IEEE Transactions on Software Engineering*, 14(8):1176–1187, August 1988.

HD77. D. W. Hammerstrom and E.S. Davidson. Information content of cpu memory referencing behavior. In *Fourth Annual Symposium in Computer Architecture*, pages 184–192, 1977.

Hob89. W.C. Hobart, Jr. *An Investigation of the Locality of Memory accesses during Symbolic Program Execution.* Ph.D. dissertation, University of Texas at Austin, August 1989.

HP90. J.L. Hennessy and D.A. Patterson. *Computer Architecture, A Quantitative Approach.* Morgan Kaufman, San Mateo, California, 1990.

Len76. J. Lenfant. Comparison of the working sets and bounded locality intervals of a program. In E. Gelenbe, editor, *Modelling and Performance Evaluation of Computer Systems*, pages 217–239. North-Holland Publishing Company, 1976.

LS73. P.A.W. Lewis and G.S. Shedler. Empirically derived micromodels for sequences of page exceptions. *IBM Journal of Research and Development*, pages 86–100, March 1973.

Mas83. Takashi Masuda. A method for the detection of program locality. In A.K. Agrawala and S.K. Tripathi, editors, *Performance '83*, pages 173–187. North-Holland Publishing Company, 1983.

MB76. A. Wayne Madison and Alan P. Batson. Characteristics of program localities. *Communications of the ACM*, 19(5):285–294, May 1976.

MB88. Jennifer M. Murphy and Richard B. Bunt. Characterising program behaviour with phases and transitions. In *Proceedings of the 1988 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pages 226–223, 1988.

MD88. Geoffrey D. McNiven and Edward S. Davidson. Analysis of memory referencing behavior for design of local memories. In *Proceedings of the 15th International Symposium on Computer Architecture*, pages 56–63, 1988.

MGST70. R.L. Mattson, J. Gecsei, D.R. Slutz, and I.L. Traiger. Evaluation techniques for storage hierarchies. *IBM Systems Journal*, 9(2):78–117, July 1970.

PD83. A.R. Pleszkun and E.S. Davidson. Structured memory access architecture. In *Proceedings of the 1983 International Conference on Parallel Processing*, pages 461–471, 1983.

SD72. Jeffrey R. Spirn and Peter J. Denning. Experiments with program locality. In *AFIPS Fall Joint Computer Conference*, pages 611–621, 1972.

SH90. Gurindar S. Sohi and Wei-Chung Hsu. The use of intermediate memories for low-latency memory access in supercomputer scalar units. *The Journal of Supercomputing*, 4(1):5–21, Mar 1990.

Sha48. C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, July 1948.

Sha50. C.E. Shannon. Predication and entropy of printed english. *Bell System Technical Journal*, 30:50–64, September 1950.

Smi87. Alan Jay Smith. Line (block) size choice for cpu cache memories. *IEEE Transactions on Computers*, C-36(9):1063–1075, September 1987.

Spi76. Jeffrey R. Spirn. Distance string models for program behavior. *IEEE Computer*, pages 14–20, November 1976.

SST88. Jaswinder Pal Singh, Harold S. Stone, and Dominique F. Thiebaut. An analytical model for fully associative cache memories. Technical Report 14232, IBM Research, T.J. Watson Research Center, 1988.

Sto87. Harold S. Stone. *High Performance Computer Architecture.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1987.

TP87. Matthew J. Thazhuthaveetil and Andrew R. Pleszkun. On the structural locality of reference in LISP list access streams. *Information Processing Letters*, 26(2):105–110, October 1987.

VMH+83. J. Voldman, B. Mandlebrot, L.W. Hoevel, J. Knight, and P. Rosenfeld. Fractal nature of software-cache interaction. *IBM Journal of Research and Development*, 27(2):164–170, March 1983.

WM88. W.S. Wong and R.J.T. Morris. Benchmark synthesis using the lru cache hit function. *IEEE Transactions on Computers*, 37(6):637–645, June 1988.

Vita

Douglas T. Michel was born in Camp Kue, Okinawa, on January 18, 1965, the son of Miyoko and Thomas Michel. After graduating from Cabrillo High School, Lompoc, California, in 1983, he entered the United States Air Force Academy, Colorado Springs, Colorado. He received the degree of Bachelor of Science in Computer Science from the United States Air Force Academy in May 1987, and was commissioned a Second Lieutenant in the United States Air Force. In May, 1991, after serving as a computer acquisition technical team leader, he entered the Air Force Institute of Technology at Wright-Patterson Air Force Base, Ohio.

Permanent address: 407 McCloud
Santa Maria CA 93455

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
December 1992

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE
A Unified Model of Program Behavior

5. FUNDING NUMBERS

6. AUTHOR(S)
Douglas T. Michel, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GCS/ENG/92D-09

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
None

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
This thesis is an attempt to account for and unify the three types of locality: temporal, spatial, and structural. A diverse sample of traces are used in measuring program behavior with respect to these localities and a model is presented which represents the memory references a program generates as it goes through execution. The model is validated by estimating the entropy of a synthetically generated trace and comparing it with actual traces. The results indicate that there is more predictability contained in the original trace than what the model was able to capture. Different variations of the model were tried and the results varied depending on the trace type being modeled. Various other measurements concerning temporal, spatial, and structural locality are used in building the model and provide interesting and useful insight into the memory referencing patterns of programs.

14. SUBJECT TERMS
Program Behavior, Locality of Reference, Cache

15. NUMBER OF PAGES
168

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL